

UNIVERSIDADE FEDERAL DE SANTA CATARINA

**PROGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA ELÉTRICA**

**RESTAURAÇÃO DA RESOLUÇÃO DE SEQUÊNCIA
DE IMAGENS USANDO REGULARIZAÇÃO
ADAPTATIVA**

Dissertação submetida à
Universidade Federal de Santa Catarina
como parte dos requisitos para a obtenção do grau de
Mestre em Engenharia Elétrica.

MARCELO VICTOR WÜST ZIBETTI

Florianópolis, Fevereiro de 2003.

RESTAURAÇÃO DA RESOLUÇÃO DE SEQUÊNCIA DE IMAGENS USANDO REGULARIZAÇÃO ADAPTATIVA

Marcelo Victor Wüst Zibetti

“Esta Dissertação foi julgada adequada para obtenção de Título de Mestre em Engenharia Elétrica, Área de Concentração *Comunicações e Processamento de Sinais*, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina”.

Joceli Mayer, Ph.D.
Orientador

Edson Roberto De Pieri, Ph.D.
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

Banca Examinadora:

Joceli Mayer, Ph.D.
Presidente da Banca

José Carlos Moreira Bermudez, Ph.D.
Membro da Banca

Leonardo Silva Resende, D.Sc.
Membro da Banca

Jefferson Luiz Brum Marques, Ph.D.
Membro da Banca

À minha família,
pelo apoio constante.

Aos meus amigos,
por acreditarem em mim.

À Juliane,
por fazer com que eu acreditasse em mim.

Agradecimentos

À Universidade Federal de Santa Catarina.

À Coordenação de Aperfeiçoamento
de Pessoal de Nível Superior – CAPES.

Ao orientador, Prof. Dr. Joceli Mayer,
pelo acompanhamento pontual e competente.

Aos professores do Curso de Pós-Graduação.

...

...

A todos que, direta ou indiretamente,
contribuíram para a realização
desta pesquisa.

Resumo da Dissertação apresentada à UFSC como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica.

RESTAURAÇÃO DA RESOLUÇÃO DE SEQUÊNCIA DE IMAGENS USANDO REGULARIZAÇÃO ADAPTATIVA

Marcelo Victor Wüst Zibetti

Fevereiro/2003

Orientador: Joceli Mayer, PhD.

Área de Concentração: Comunicações e Processamento de Sinais.

Palavras-chave: interpolação, aquisição de imagens, restauração iterativa, sequência de imagens, regularização adaptativa.

Número de Páginas: 103.

Esta pesquisa apresenta o processo de restauração da resolução de imagens adquiridas por sistemas de captura de imagens. O processo faz uso da informação presente em cada quadro em uma sequência de imagens. Os modelos de aquisição de imagens e os modelos de movimento entre as imagens são investigados. A técnica iterativa de Mínimos Quadrados Regularizados é utilizada e desenvolvida para a aplicação em sequências de imagens com *outliers**. O trabalho enfatiza que imagens com movimento de *subpixel** trazem novas informações e contribuição para a restauração da resolução. Esse tipo de informação é capaz de evitar distorções causadas pela subamostragem e pode recuperar os detalhes perdidos durante a aquisição. Considerando que o movimento entre as imagens adquiridas não pode ser controlado, a contribuição das imagens adicionais não pode ser controlada e é dispersa na imagem restaurada. Além disso, a restauração é prejudicada pela presença dos *outliers*. Este trabalho propõe uma regularização adaptativa para minimizar as distorções causadas pela subamostragem, suavizando as regiões com pouca contribuição e permitindo a recuperação de detalhes nas regiões de contribuição significativa. Resultados experimentais apresentados neste trabalho confirmam que a regularização adaptativa traz benefícios objetivos (melhora da relação sinal/ruído) e subjetivos (melhoria de qualidade perceptual).

* Ver significado no Glossário.

Abstract of Dissertation presented to UFSC as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering.

RESOLUTION RESTORATION OF IMAGE SEQUENCES USING ADAPTIVE REGULARIZATION

Marcelo Victor Wüst Zibetti

February/2003

Advisor: Joceli Mayer, PhD.

Area of Concentration: Communications and Signal Processing.

Keywords: interpolation, image acquisition, iterative restoration, image sequences, adaptive regularization.

Number of Pages: 103.

This research presents a process of resolution restoration of images captured by an image acquisition system. The process considers the contribution of each image in the image sequence. The acquisition models and the motion models among the images are investigated. The iterative Regularized Least-Squares technique is presented as a solution and developed to be applied in sequences with outliers. During the investigation we found that the images with subpixel displacements can provide new information and contribute to the resolution restoration. Subpixel information can be used to mitigate the distortions caused by sub-sampling process and to recover the details lost during the acquisition. Assuming that motion between the acquired images cannot be controlled, then the subpixel contribution cannot be controlled as well and are dispersed in the estimated image. Therefore, the contributions are reduced by the presence of outliers. Then, this work proposes an adaptive regularization to mitigate the distortions caused by the sub-sampling, smoothing the regions with little contribution and allowing the recovery of the details in the regions of significant contribution. Experimental results indicate that the adaptive regularization provides objective (signal-to-noise-ratio improvement) and subjective benefits (perceptual quality improvement).

Sumário

Capítulo 1: Introdução.....	1
1.1. Justificativa	1
1.2. Seqüência de Imagens e Recuperação da Degradação de Aquisição	2
1.3. Motivação do Trabalho.....	4
1.4. Organização do Trabalho.....	4
1.5. Assuntos Apresentados	5
Capítulo 2: Modelo de Degradação	6
2.1. Modelo Contínuo-Discreto	6
2.2. Modelo Discreto-Discreto	7
2.3. Degradação Usando Notação Vetorial	9
2.4. Componentes de Degradação	10
2.5. Exemplo Prático.....	12
Conclusões do Capítulo 2	15
Capítulo 3: Modelos de Movimento	16
3.1. Movimento em Imagens	16
3.2. Composição do Modelo de Movimento.....	18
3.3. Modelo de Movimento com <i>Inliers</i> e <i>Outliers</i>	22
Conclusões do Capítulo 3	25
Capítulo 4: Soluções para Inverter a Degradação.....	26
4.1. Análise da Solução de Sistemas Lineares	26
4.2. Solução por Otimização e Regularização	28
4.3. Seqüência de Imagens e Conjuntos Teóricos.....	34
Conclusões do Capítulo 4	39
Capítulo 5: Solução Iterativa Regularizada	40
5.1. Soluções Iterativas de Projeção de Gradiente	40
5.2. Propriedades Computacionais do Método Iterativo.....	45
Conclusões do Capítulo 5	50
Capítulo 6: Estimação de Movimento Entre as Imagens.....	51
6.1. Estimação de Movimento Via Combinação de Intensidade.....	51
6.2. Estimação de Movimento de Imagens de Alta Resolução a partir de Imagens de Baixa Resolução	53
6.3. Estimando Movimento com <i>Inliers</i> e <i>Outliers</i>	56

6.4.	Estimação de Movimento e de <i>Inliers</i> e <i>Outliers</i> a Partir de Imagens de Baixa Resolução	57
	Conclusões do Capítulo 6	59
Capítulo 7:	Solução Iterativa de Sequência de Imagens com <i>Inliers</i>	60
7.1.	Modelo de Degradação Considerando <i>Inliers</i> e <i>Outliers</i>	60
7.2.	Mudança no Equacionamento da Solução Iterativa	61
7.3.	Compensação na Projeção do Método Iterativo com <i>Inliers</i>	63
	Conclusão do Capítulo 7	65
Capítulo 8:	Regularização Adaptativa para Sequência de Imagens	66
8.1.	Regularização das Distorções Causadas pela Subamostragem	66
8.2.	Sequência de Imagens e Recuperação dos Efeitos da Subamostragem	69
8.3.	Uso do Modelo com <i>Inliers</i> e <i>Outliers</i>	70
8.4.	Regularização Adaptativa para os Efeitos da Subamostragem	71
8.5.	Descrição do Processo Completo	75
8.6.	Acréscimo de Custo Computacional	76
	Conclusões do Capítulo 8	77
Capítulo 9:	Resultados Experimentais Simulados	78
9.1.	Simulação 1	78
9.2.	Simulação 2	85
	Conclusões do Capítulo 9	89
Capítulo 10:	Aplicações em Casos Reais	90
10.1.	Zoom de Uma Cena Estática	90
10.2.	Melhoria de Definição de Textos	91
	Conclusões do Capítulo 10	92
Conclusões e Discussões Futuras	93
	Conclusões Finais	93
	Discussões Futuras	94
Apêndice A1:	Subamostragem Regular	96
A1.1.	Caso Unidimensional, Subamostragem de Duas Vezes	96
A1.2.	Caso Bidimensional, Subamostragem de Duas Vezes	97
Referências Bibliográficas	99
Glossário	102

Listas

1. Figuras

Figura 1.1: Demonstração de quantização e amostragem.	1
Figura 1.2: Sinais unidimensionais sob subamostragem.	3
Figura 1.3: Etapas envolvidas do processo.	4
Figura 2.1: Diagrama de blocos do modelo de degradação Contínuo-Discreto.	6
Figura 2.2: Demonstração das etapas do processo de degradação.	9
Figura 2.3: Degradação aplicada a cada uma das L imagens e o movimento entre elas.	10
Figura 2.4: Grade representativa de subamostragem do CCD utilizado.	14
Figura 3.1: Trajetória de movimento do <i>pixel</i>	17
Figura 3.2: Regiões de movimento.	19
Figura 3.3: Exemplos de tipos de movimento.	21
Figura 3.4: <i>Aliasing</i> na formação de uma das imagens com movimento.	22
Figura 3.5: Regiões de <i>outliers</i> entre as imagens marcadas em preto.	25
Figura 4.1: Composição dos quatro quadros LR na grade HR para reconstrução.	28
Figura 4.2: Valores singulares e perturbação devido ao ruído.	31
Figura 4.3: Representação das elipsóides para casos bidimensionais.	32
Figura 4.4: Representação do elipsóide quando D_m é 1×2 , C é 2×2	33
Figura 4.5: Representação das funções objetivo através do uso de elipsóides.	36
Figura 4.6: Elipsóides geradas pelas imagens em combinação com a restrição.	38
Figura 5.1: Busca pelo mínimo no caso Descida mais Íngreme.	44
Figura 5.2: Comparação de custo computacional entre os métodos.	50
Figura 6.1: Corte unidimensional de funções objetivo usando as imagens originais e as imagens estimadas com interpolação bicúbica.	55
Figura 7.1: Problema no uso de seqüência de imagens no processo iterativo.	64
Figura 7.2: Efeito na imagem da compensação na projeção do método iterativo.	65
Figura 8.1: Aparecimento de componentes de alta frequência devido à subamostragem.	67
Figura 8.2: Exemplo do espectro de um filtro $G^T G$ escolhido como restrição.	68
Figura 8.3: Exemplo de imagens estimadas com distorções da subamostragem e regularização.	69
Figura 8.4: Inversão dos efeitos da subamostragem através do uso de seqüência de imagens.	70
Figura 8.5: Exemplo de inversão dos efeitos da subamostragem de forma local.	71
Figura 8.6: Espectro dos filtros da regularização adaptativa.	72
Figura 8.7: Exemplo da análise dos vetores para determinação da contribuição de <i>subpixel</i>	73
Figura 8.8: Etapas de processamento e análise anteriores ao processo iterativo.	75

Figura 8.9: Diagrama de blocos do processo iterativo	75
Figura 8.10: Custo computacional em relação ao tamanho das imagens.....	77
Figura 9.1: Imagens da seqüência de alta resolução.....	79
Figura 9.2: Imagens da seqüência degradada artificialmente.....	79
Figura 9.3: Superfície formada pela norma quadrática do erro de estimação.	80
Figura 9.4: Estimação dos <i>outliers</i>	80
Figura 9.5: Espectro dos filtros G_i	81
Figura 9.6: Pesos W_i dos filtros G_i	82
Figura 9.7: Resultados visuais das técnicas.	84
Figura 9.8: Imagens da seqüência de alta resolução.....	85
Figura 9.9: Seqüência degradada.	86
Figura 9.10: Resultados visuais das técnicas.	88
Figura 10.1: Recuperação da resolução de duas vezes.....	91
Figura 10.2: Ampliação de duas vezes.....	92
Figura A.1.1: Sinal ao sofrer subamostragem.....	96
Figura A.1.2: Sinais bidimensionais subamostrados.	98

2. Tabelas

Tabela 3.1: Tipos de movimento.....	20
Tabela 5.1:Diferença de custo computacional	49
Tabela 8.1: Tipos de operadores.	68
Tabela 8.2: Tabela com acréscimo de custo computacional.	76
Tabela 9.1: Comparação dos resultados.....	83
Tabela 9.2: Comparação o dos resultados.....	87

3. Siglas

2D	– <i>2-Dimensional</i> , ou Bidimensional.
3D	– <i>3-Dimensional</i> , ou Tridimensional.
BEM	– <i>Bounding Ellipsoidal Method</i> , ou Método de Elipsóide de Contorno.
CCD	– <i>Charge Couple Device</i> , um tipo de sensor eletrônico de imagem.
CLS	– <i>Constrained Least Squares</i> , ou Mínimos Quadrados Restritos.
DFD	– <i>Displaced Frame Difference</i> , ou Diferença entre Quadros Deslocados.
DSNR	– <i>Differential Signal-to-Noise Ratio</i> , ou Relação Sinal/Ruído Diferencial.
FFT	– <i>Fast Fourier Transform</i> , ou Transformada Rápida de Fourier.
FLOPS	– <i>Float Point Operations</i> , ou Operações de Ponto Flutuante.

HR	– <i>High Resolution</i> , ou Alta Resolução.
LR	– <i>Low Resolution</i> , ou Baixa Resolução.
LS	– <i>Least Squares</i> , ou Mínimos Quadrados.
LSI	– <i>Linear Space Invariant</i> , ou Linear Espaço Invariante.
MA	– <i>Moving Average</i> , ou Média Móvel.
PSF	– <i>Point Spread Function</i> , ou Função de Espalhamento de Ponto.
PSNR	– <i>Peak Signal-to-Noise Ratio</i> , ou Relação Sinal/Ruído de Pico.
RLS	– <i>Regularized Least Squares</i> , ou Mínimos Quadrados Regularizados.
SVD	– <i>Singular Value Decomposition</i> , ou Decomposição em Valores Singulares.
TLS	– <i>Total Least Squares</i> , ou Mínimos Quadrados Totais.
WLS	– <i>Weighted Least Squares</i> , ou Mínimos Quadrados Ponderados.

4. Notação

$\mathbf{D}, \mathbf{D}_{k,m}$	(maiúsculo, negrito)	Matrizes em geral.
\mathbf{D}, \mathbf{D}_m	(maiúsculo, itálico, negrito)	Matrizes formadas por concatenação.
$\hat{\mathbf{A}}_{k,m}$	(maiúsculo, negrito, com “^”)	Matrizes estimadas.
$\mathbf{f}, \mathbf{f}_m, \mathbf{g}_k^I$	(minúsculo, negrito)	Vetores.
\mathbf{g}	(minúsculo, itálico, negrito)	Vetores formados por concatenação.
$\hat{\mathbf{e}}_{k,m}$	(minúsculo, negrito, com “^”)	Vetores estimados.
x	(minúsculo, itálico)	Escalares.
\hat{e}	(minúsculo, itálico, com “^”)	Escalares estimadas.
M_1, R	(maiúsculo)	Constantes escalares.
$Y(z), f_m(x)$	(itálico, com parênteses)	Funções escalares contínuas.
$\mathbf{e}_{k,m}(x)$	(negrito, com parênteses)	Funções vetoriais contínuas.
$f_m[m]$	(itálico, com colchetes)	Funções escalares discretas.
$\mathbf{v}_{k,m}[m]$	(negrito, com colchetes)	Funções vetoriais discretas.
$e_{k,m}(\boldsymbol{\theta})$	(itálico, com variável vetor)	Função escalar contínua, com múltiplas variáveis (variável vetor).
$\mathbf{v}_{k,m}(\mathbf{m})$	(minúsculo, negrito, “()”, com vetor)	Função vetorial contínua, com múltiplas variáveis (variável vetor).
$\mathbf{v}_{k,m}[\mathbf{m}]$	(minúsculo, negrito, “[]”, com vetor)	Função vetorial discreta, com múltiplas variáveis (variável vetor).
$\mathbf{A}_{k,m}(\boldsymbol{\theta})$	(maiúsculo, negrito, “()”, com vetor)	Função matriz contínua, com múltiplas variáveis (variável vetor).

Capítulo 1: Introdução

Em diversas aplicações o processamento digital é extremamente necessário para melhorar a resolução de um sinal. Melhorar a resolução de um sinal digital significa conseguir uma melhor aproximação do sinal contínuo, seja qual for sua natureza. O termo *resolução* implica dois pontos: quantização e amostragem. Quando o sinal desejado é uma imagem, tem-se que a *quantização* implica a quantidade de níveis ou tons de iluminação que a representação digital pode atingir; já a *amostragem* implica a quantidade de amostras espaciais, pontos ou *pixels*, necessárias para representar determinada região da imagem. Observem-se essas descrições na Figura 1.1.

Esta pesquisa apresenta uma forma de melhorar a resolução quanto à *amostragem*, sugerindo formas de aplicação do processamento digital de modo a se obter uma imagem com taxa de amostragem aparente maior que a fornecida pelo sistema de aquisição. Por se tratar a *melhoria de resolução* através de modelos matemáticos, que consideram a degradação na imagem, pode-se considerar o assunto como *restauração da resolução*.

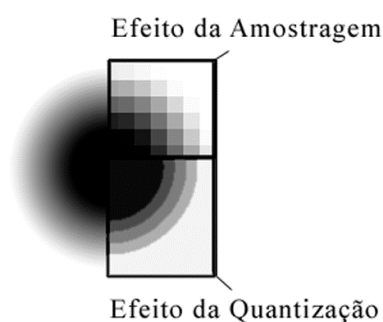


Figura 1.1: Demonstração de quantização e amostragem.

1.1. Justificativa

Imagens de alta resolução (HR - *High-Resolution*) são ferramentas úteis em várias aplicações, desde televisão de alta definição, multimídia, imagens médicas e científicas, fotos aéreas e de satélites, astronomia, até uso militar. No entanto, os sistemas eletrônicos de captura de imagem possuem um limite físico de resolução de imagem. Melhorar a resolução através do uso de fotossensores mais densos, lentes mais limpas e sistemas de foco mais precisos pode ser custoso ou até impossível em alguns casos. Uma das aproximações possíveis para a solução do problema é a restauração através do uso de sequência de imagens.

Embora muitas técnicas de interpolação tenham sido pesquisadas e desenvolvidas, a maioria delas não aproveita toda a informação disponível, como o modelo de degradação do dispositivo de captura e a diferença de informação entre os vários quadros (*frames*) de uma sequência de imagens. Os métodos de interpolação convencionais, como as interpolações de Ordem Zero, Bilinear e

Bicúbica, são consideradas funções básicas e apenas aumentam a imagem de baixa resolução (LR - *Low-Resolution*) sem corrigir possíveis degradações. Para se obter melhores resultados, é necessário considerar a ocorrência de mistura entre os *pixels* adjacentes no sensor de imagem, a distorção por movimento (*motion blur*) devido ao tempo de exposição necessário para a captura da imagem, a distorção isotrópica devido à falta de foco e à sobreposição de espectro (*aliasing*) no processo de subamostragem. As técnicas de restauração da resolução através do uso de sequência de imagens são uma alternativa para recuperar os detalhes perdidos na subamostragem, dado que cada imagem possui uma nova e exclusiva informação sobre a imagem de alta resolução. Essa diferença de informação é gerada por objetos na sequência de imagens que se movem com incrementos de *subpixel* e devido às diferenças na degradação de cada imagem da sequência. Então, a restauração de resolução aproveita as informações contidas na sequência de imagens capturadas e as informações da degradação do dispositivo de aquisição para estimar uma imagem de alta resolução com maior qualidade.

1.2. Sequência de Imagens e Recuperação da Degradação de Aquisição

Todos os sistemas de aquisição de imagem digital têm uma capacidade limite de aproximação da imagem contínua original pela imagem digital. A imagem digital capturada pode ser vista como uma versão degradada da imagem original. Então, pode-se considerar o sistema de aquisição como um sistema que causa uma degradação na imagem original. É necessário que a degradação do sistema seja conhecida, ou aproximada, para que seus efeitos possam ser removidos. Dado o conhecimento da imagem degradada e do sistema de captura, pode-se tentar encontrar a solução inversa do problema. O modelo de degradação de um sistema de aquisição geral será apresentado no Capítulo 2.

Todo modelo de aquisição de imagem digital respeita os princípios gerais de amostragem, sendo, assim, formado basicamente por um filtro passa-baixa e um amostrador. A vantagem do uso de sequência de imagens surge quando o filtro do sistema não é ideal, o que ocorre nos casos reais. Assim, a mais alta frequência da imagem real acaba sendo amostrada em taxa menor que a de Nyquist, causando *aliasing* no espectro da imagem formada. Dessa forma, além de não conseguir adquirir as altas frequências, as baixas frequências amostradas também são prejudicadas. Uma forma de se recuperar a imagem original e remover os efeitos do *aliasing* é a obtenção das amostras dizimadas no processo de subamostragem. Com o movimento espacial de *subpixel* é possível se obter isso.

Representa-se, agora, tomando por base um exemplo unidimensional, o modo como o uso das imagens, com movimento de *subpixel*, pode recuperar o sinal original, removendo os efeitos do *aliasing*. O sinal $y[n]$ apresentado na Figura 1.2.a seria o sinal capturado por um sistema de

aquisição ideal, com filtro passa-baixa ideal e amostragem acima da taxa de Nyquist, que é a taxa de amostragem suficiente.

Supondo que o sistema tivesse uma taxa de amostragem duas vezes menor que a suficiente, o sinal $x_1[n]$, demonstrado na Figura 1.2.b, é o primeiro sinal de uma seqüência de sinais adquiridos da observação de $y[n]$. O sinal $x_2[n]$, representado na Figura 1.2.c, é o segundo sinal da seqüência e possui movimento de *subpixel* de $\frac{1}{2}$ *pixel* em relação a $x_1[n]$.

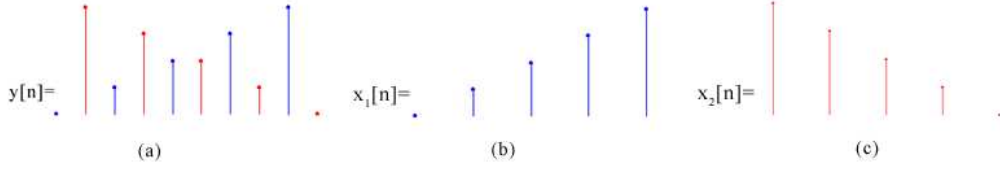


Figura 1.2: Sinais unidimensionais sob subamostragem.

Para se recuperar o sinal $y[n]$, é necessário conhecer a seqüência de sinais composta dos sinais $x_1[n]$, $x_2[n]$ e o movimento de *subpixel* exato entre eles. Matematicamente, pode-se demonstrar que o espectro de $x_1[n]$ e $x_2[n]$ no Plano Z é:

$$\begin{aligned} X_1(z) &= \frac{1}{2}[Y(z^{1/2}) + Y(-z^{1/2})] \\ X_2(z) &= \frac{1}{2}[z^{1/2}Y(z^{1/2}) - z^{1/2}Y(-z^{1/2})] \end{aligned} \quad (1.1)$$

Para recuperar o sinal $y[n]$, é necessário realizar a seguinte operação:

$$\begin{aligned} Y(z) &= X_1(z^2) + z^{-1}X_2(z^2) \\ Y(z) &= \frac{1}{2}[Y(z) + Y(-z)] + z^{-1}\frac{1}{2}[zY(z) - zY(-z)] \\ Y(z) &= \frac{1}{2}[Y(z) + Y(-z)] + \frac{1}{2}[Y(z) - Y(-z)] \end{aligned} \quad (1.2)$$

Para deixar as Equações (1.1) e (1.2) mais claras, considera-se que $z^{1/2}$ equivale à redução da taxa de amostragem por dois (*downsampling*) e z^2 equivale ao aumento da taxa de amostragem (*upsampling*) por dois. O termo $Y(-z^{1/2})$ em $X_1(z)$ e $X_2(z)$ demonstra o aparecimento da versão de alta frequência de $Y(z)$ durante o processo de subamostragem (aproximação dos espectros da frequência zero). O termo $z^{1/2}$ multiplicando $Y(z^{1/2})$ e $-z^{1/2}$ multiplicando $Y(-z^{1/2})$ em $X_2(z)$ demonstra o deslocamento de *subpixel*. Deduções completas sobre essa parte são encontradas no Apêndice.

Essa possibilidade vem motivando muitos trabalhos de recuperação das altas frequências e eliminação de sobreposição de espectro (*aliasing*) de sinais amostrados, principalmente de imagens. Essa é uma forma de se obterem sinais amostrados com taxa de amostragem maior que a taxa que o amostrador convencional permite obter.

1.3. Motivação do Trabalho

Este trabalho é desenvolvido sob a condição de que o uso de múltiplas imagens, ou sequência de imagens, pode trazer resultados, quanto à restauração da resolução, que o uso de uma única imagem não é capaz de conseguir. Trabalhos já desenvolvidos, referenciados ao longo dos capítulos subseqüentes e ainda em desenvolvimento na área, demonstraram que esta proposta é viável.

Baseando-se na teoria já desenvolvida até aqui por trabalhos anteriores, observou-se que ainda há muitos pontos a serem explorados. Este trabalho apresenta as linhas mais bem-sucedidas e propõe alguns melhoramentos.

1.4. Organização do Trabalho

O processo de degradação envolvido na geração de imagem pelo dispositivo de aquisição e a tentativa de restauração estão representados de forma simplificada na Figura 1.3. As etapas envolvidas serão tratadas nos capítulos subseqüentes.

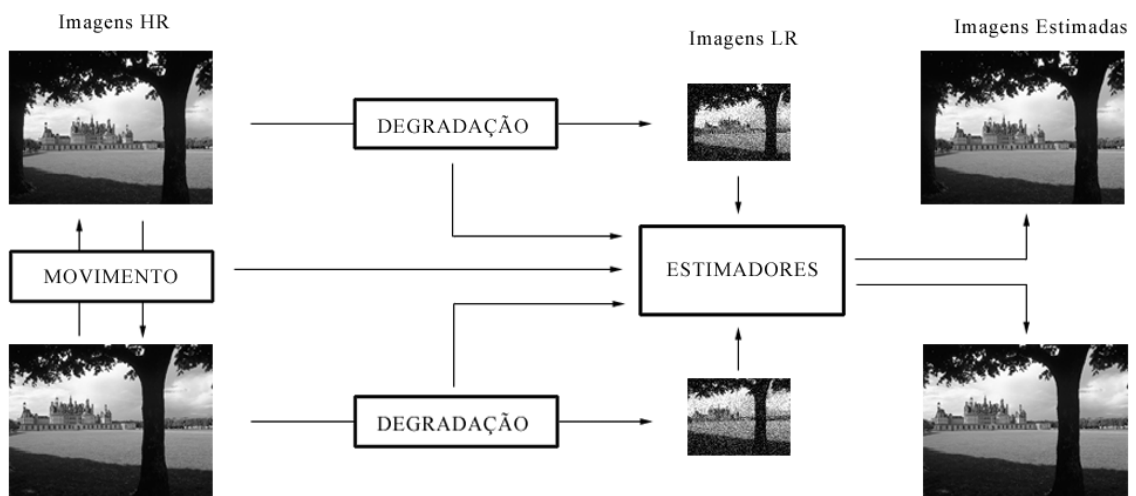


Figura 1.3: Etapas envolvidas do processo.

No Capítulo 2 é apresentado o modelo de degradação, representado pelo bloco DEGRADAÇÃO da Figura 1.3. No Capítulo 3 são tratados os modelos de movimento envolvidos, que estão representados pelo bloco MOVIMENTO. Os Capítulos 4, 5 e 6 tratam das soluções para os estimadores, representados pelo bloco ESTIMADORES da Figura 1.3. Estes capítulos abordam o problema da estimação de movimento (se este não for conhecido) e apresentam soluções para inversão da degradação. Nos Capítulos 7 e 8 são tratados as novas propostas de solução do problema e o melhoramento das técnicas apresentadas. Os Capítulos 9 e 10 apresentam resultados experimentais simulados e práticos.

1.5. Assuntos Apresentados

Esta seção foi introduzida para facilitar a localização dos assuntos ao longo dos textos da dissertação. Será feita uma breve descrição dos assuntos para orientar melhor o leitor já familiarizado com o tema e com os conteúdos.

No Capítulo 2, apresenta-se o modelo de degradação dos sistemas de aquisição. O modelo envolve movimento, degradação óptica, subamostragem e ruído. Cada componente é explicado. A degradação é formulada, para uso posterior, através de notação vetorial.

No Capítulo 3, o movimento entre as imagens é explicado. As imagens são relacionadas através de equações com vetores de movimento e através de transformação linear de compensação de movimento. A presença de *outliers* e o tratamento matemático dos mesmos são demonstrados.

No Capítulo 4, a solução para inversão da degradação é apresentada como um problema de inversão de matrizes mal-condicionadas. A solução de Mínimos Quadrados Regularizados é apresentada, e é demonstrado o uso de conjuntos teóricos para múltiplas imagens.

No Capítulo 5, a busca da solução através de iteração é demonstrada. São apresentadas as iterações de Distanciamento Simultâneo (*Simultaneous Displacement*) e de Decida Mais Íngreme (*Steepest Descent*). As vantagens do método iterativo são explicadas.

No Capítulo 6, são apresentados os problemas de estimação de movimento, usando as técnicas de combinação de intensidade quando as imagens usadas são de baixa resolução. Os problemas na estimação dos *outliers* são explicados.

No Capítulo 7, a técnica iterativa regularizada com sequência de imagens, tradicionalmente encontrada na literatura, é modificada para uso em sequência com *inliers*. Um novo modelo de degradação é formulado. Novas sugestões para determinação dos fatores de regularização são apresentadas, e o uso de uma compensação de projeção para o método iterativo é proposto.

No Capítulo 8, a principal contribuição é apresentada, sendo proposta a regularização adaptativa para minimizar as distorções causadas pela subamostragem. A contribuição das imagens da sequência, com informação de *subpixel*, para a minimização das distorções da subamostragem e recuperação dos detalhes é explicada.

Nos Capítulos 9 e 10, simulações e aplicações práticas das técnicas propostas ao longo desta dissertação são apresentadas para comprovar a teoria.

As conclusões finais e discussões para trabalhos futuros são encontradas ao final da dissertação.

No Apêndice constam as informações sobre a subamostragem e as comprovações matemáticas da vantagem do uso de imagens com movimento de *subpixel*.

Capítulo 2: Modelo de Degradação

Neste capítulo, apresenta-se o modelo de degradação utilizado neste trabalho. O modelo de degradação do sistema de aquisição de imagem é uma informação essencial para a restauração, sendo possível, através dele, relacionar as imagens de alta resolução com as imagens de baixa resolução.

O modelo de degradação será, inicialmente, apresentado da forma contínua para discreta, como é em sua natureza, e, em seguida, na forma discreta de alta resolução para discreta de baixa resolução, que é a abordagem com que ele será tratado utilizando processamento digital. Este modelo é derivado dos modelos de degradação encontrados na literatura [1,2,10-12].

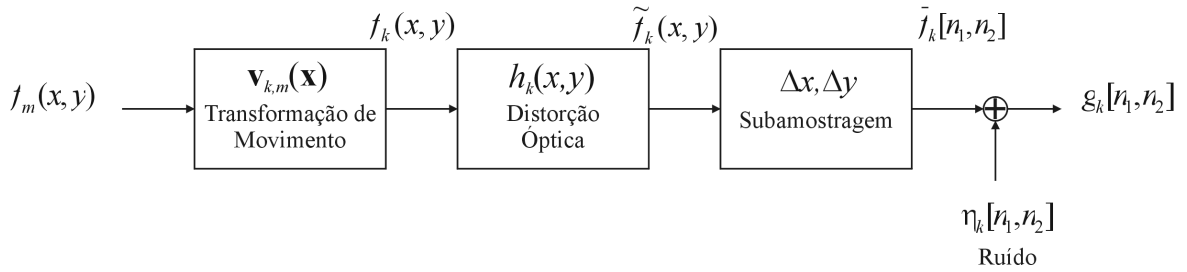


Figura 2.1: Diagrama de blocos do modelo de degradação Contínuo-Discreto.

2.1. Modelo Contínuo-Discreto

Assume-se que um objeto tridimensional (3D), ou uma cena, é projetado sobre um plano bidimensional (2D) em um determinado instante e posição espacial por um dispositivo de aquisição. Um diagrama de blocos do modelo de degradação é mostrado na Figura 2.1. A imagem original contínua que está sob observação é chamada $f_m(x,y)$ no sistema de coordenadas 2D contínuas (x,y) . Essa imagem de alta resolução sofre várias degradações, representadas pelos blocos transformação de movimento, distorção óptica (*blurring*), subamostragem e ruído, antes de ser fornecida pelo sistema.

A primeira etapa é a transformação de movimento e descreve o movimento espacial entre posições relativas das imagens. Movimentos como translação, rotação, escalamento e movimento de objetos em posições aleatórias são modelados para demonstrar o movimento da imagem contínua m para a imagem contínua k . As características da transformação de movimento estão de acordo com o modelo de movimento, apresentado no Capítulo 3.

Conforme o modelo de aquisição, a relação do k -ésimo quadro de imagem observado na sequência com a imagem de interesse m pode ser expressa como:

$$f_k(\mathbf{x}) = f_m(\mathbf{x} + \mathbf{v}_{k,m}(\mathbf{x})) \quad \text{para } k = 1, 2, \dots, L, \quad (2.1)$$

onde $\mathbf{v}_{k,m}(\mathbf{x})$ representam os vetores da transformação de movimento nas coordenadas $\mathbf{x}=(x,y)$, e onde a cada transformação $m \rightarrow k$ a imagem na posição contínua (x,y) em m move-se para posição diferente em k . Os índices k e m podem ser vistos como o instante no tempo, se for uma sequência de vídeo, ou o ponto de vista da cena, se for uma sequência adquirida por várias câmeras.

O efeito de distorção óptica causada pelos fotodetectores, difração no sistema de lentes e outras distorções do fluxo de luz é modelado como o segundo passo da Figura 2.1 através da convolução com o núcleo de distorção, $h_k(x,y)$:

$$\tilde{f}_k(x,y) = f_k(x,y) * h_k(x,y), \quad (2.2)$$

Nota-se que esta distorção é assumida como sendo linear e espacialmente invariante (LSI – *Linear Space Invariant*) em (2.2), e, ainda, pode ser diferente em cada imagem k , seja porque é temporalmente variante (se for vídeo), seja porque é diferente em cada câmera que está sendo utilizada (se for sequência adquirida por várias câmeras).

O terceiro passo do modelo de degradação é a subamostragem. Em geral, a grade de amostragem é retangular 2D, mas também pode ser hexagonal ou outro tipo. Por isso, a versão subamostrada de (2.2) pode ser obtida como:

$$\bar{f}_k(x,y) = \tilde{f}_k(x,y) p_k(x,y), \quad (2.3a)$$

onde a função $p_k(x,y)$ modela a grade de amostragem. Como exemplo, no caso de grade retangular 2D, tem-se o modelo da Equação (2.3b).

$$p_k(x,y) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \delta(x - n_1 \Delta x, y - n_2 \Delta y), \quad (2.3b)$$

onde $\delta(x,y)$ representa a função delta de Dirac [1,6].

A grade retangular leva à seguinte relação do sinal contínuo com o sinal discreto:

$$\bar{f}_k[n_1, n_2] = \bar{f}_k(x,y), \quad \text{quando } x = n_1 \Delta x \text{ e } y = n_2 \Delta y \text{ e } n_1 = 0,1,\dots, N_1, \quad n_2 = 0,1,\dots, N_2 \quad (2.4)$$

Finalmente, a imagem discreta com transformação de movimento, distorção óptica e subamostrada, é corrompida por ruído aditivo, conforme:

$$g_k[n_1, n_2] = \bar{f}_k[n_1, n_2] + \eta_k[n_1, n_2] \quad (2.5)$$

O ruído é modelado como um processo aleatório gaussiano e estatisticamente independente da imagem. Cada componente da degradação, bem como as considerações sobre sua origem e determinação são detalhadas na Seção 2.4.

2.2. Modelo Discreto-Discreto

Aqui, as imagens de baixa resolução amostradas são expressas em termos das imagens de alta resolução amostradas. O problema de restauração de resolução é formulado como o de

reconstrução da imagem de alta resolução a partir de uma seqüência de imagens de baixa resolução. Assume-se que a imagem contínua de alta resolução, $f_m(x,y)$, é amostrada acima da taxa de Nyquist, então, sua versão amostrada não tem *aliasing*. Pela amostragem ideal, acima da taxa de Nyquist, a imagem discreta de alta resolução, $f_m[m_1, m_2]$, pode ser produzida. Considera-se o tamanho da imagem discreta de alta resolução como $M_1 \times M_2$.

Como na seção anterior, a imagem discreta de alta resolução, $f_m[m_1, m_2]$, sofre os mesmos passos de degradação descritos na Figura 2.1, porém na forma discreta. Primeiramente, a transformação de movimento:

$$f_k[\mathbf{m}] = f_m[\mathbf{m} + \mathbf{v}_{k,m}[\mathbf{m}]] \quad \text{para} \quad k=1, \dots, L, \quad (2.6)$$

onde $\mathbf{v}_{k,m}[\mathbf{m}]$ representam os vetores da transformação de movimento discretos nas coordenadas discretas de alta resolução $\mathbf{m}=[m_1, m_2]$.

Então, contando com o efeito da distorção óptica dos fotodetectores, lentes e demais responsáveis pelo fluxo da luz, a função discreta de distorção, $h_k[m_1, m_2]$, é modelada como uma função de espalhamento de ponto (PSF – *Point Spread Function*) e está descrita abaixo:

$$\tilde{f}_k[m_1, m_2] = f_k[m_1, m_2] * h_k[m_1, m_2] \quad (2.7)$$

Depois, a imagem que sofreu transformação de movimento e distorção óptica é subamostrada:

$$\bar{f}_k[m_1, m_2] = \tilde{f}_k[m_1, m_2] p_k[m_1, m_2], \quad (2.8)$$

onde $p_k[m_1, m_2]$ é a função que representa a dizimação da grade de alta resolução. A relação de subamostragem resulta em:

$$\bar{f}_k^{\downarrow \Delta m_1, \Delta m_2}[n_1, n_2] = \bar{f}_k[m_1, m_2], \quad \text{quando} \quad m_1 = n_1 \Delta m_1 \quad \text{e} \quad m_2 = n_2 \Delta m_2, \quad (2.9)$$

onde $\bar{f}_k[m_1, m_2]$ tem tamanho $M_1 \times M_2$ e $\bar{f}_k^{\downarrow \Delta m_1, \Delta m_2}[n_1, n_2]$ tem tamanho $N_1 \times N_2$.

Finalmente, a imagem discreta subamostrada, $\bar{f}_k^{\downarrow \Delta m_1, \Delta m_2}[n_1, n_2]$, é corrompida por ruído aditivo como:

$$g_k[n_1, n_2] = \bar{f}_k^{\downarrow \Delta m_1, \Delta m_2}[n_1, n_2] + \eta_k[n_1, n_2], \quad (2.10)$$

onde o ruído, que é o mesmo apresentado na seção anterior, é um processo aleatório gaussiano e independente da imagem.

Como resultado, são observadas L imagens de baixa resolução, $\{g_k[n_1, n_2]\}$ onde $k=1 \dots L$, produzidas pela seqüência de imagens de alta resolução, $\{f_k[m_1, m_2]\}$ onde $k=1 \dots L$. Todas as imagens estão relacionadas com a imagem $f_m[m_1, m_2]$, que faz parte da seqüência de alta resolução e que se pretende recuperar a partir da seqüência degradada.

A Figura 2.2 demonstra os processos envolvidos na degradação de uma forma simplificada.

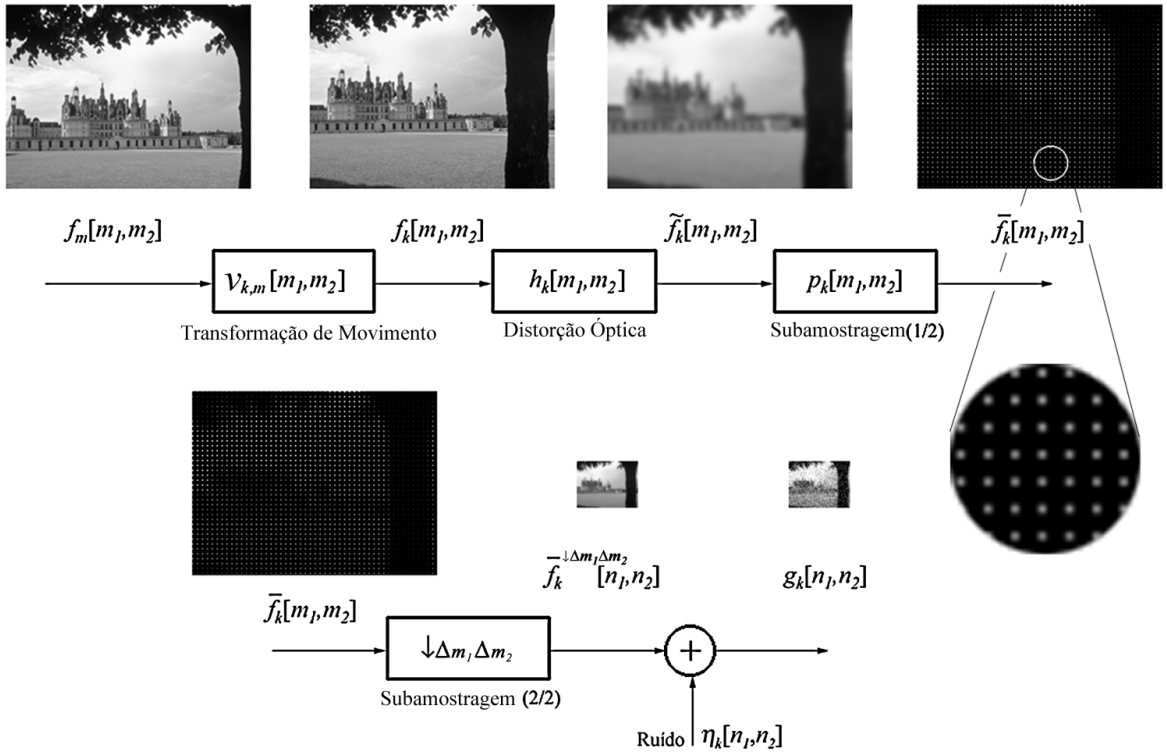
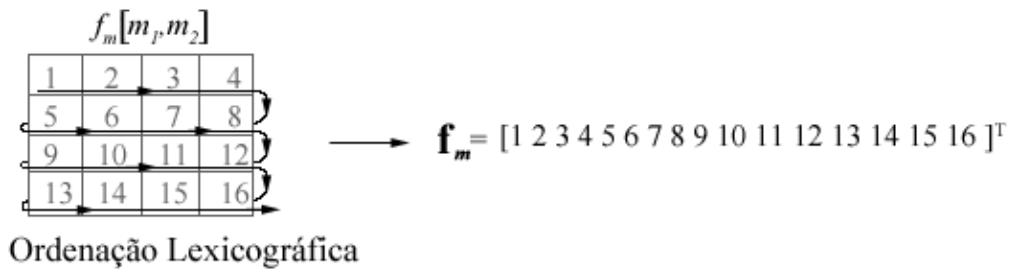


Figura 2.2: Demonstração das etapas do processo de degradação.

2.3. Degradação Usando Notação Vetorial

Os processos envolvidos no modelo de degradação são complexos e sua resolução é de difícil manipulação na forma como foi apresentada. Por isso, torna-se preferencial o uso da notação vetorial ou matriz-vetor, que é possível dado que as operações são lineares.

Assim, a imagem $f_m[m_1, m_2]$ é reordenada lexicograficamente e compõe o vetor \mathbf{f}_m , que tem dimensões $M_1 M_2 \times 1$. De forma semelhante, as operações lineares passam a ser matrizes de transformação.



O processo pode ser representado, em notação vetorial, como:

$$\mathbf{g}_k = \mathbf{S}_k \mathbf{H}_k \mathbf{A}_{k,m} \mathbf{f}_m + \boldsymbol{\eta}_k, \quad (2.11)$$

onde $\mathbf{A}_{k,m}$ é uma matriz $M_1 M_2 \times M_1 M_2$ que representa a transformação de movimento. A matriz \mathbf{H}_k , de tamanho $M_1 M_2 \times M_1 M_2$, representa a distorção óptica. A matriz \mathbf{S}_k , de tamanho $N_1 N_2 \times M_1 M_2$, representa a subamostragem. O ruído é modelado como o vetor $\boldsymbol{\eta}_k$ de tamanho $N_1 N_2 \times 1$.

O fator de ampliação (R), que determina a diferença entre a resolução desejada e a resolução do sistema de aquisição, pode ser expresso como: $M_1 M_2 = R^2 N_1 N_2$, onde $R^2 = \Delta m_1 \Delta m_2$.

O processo também pode ser tratado como:

$$\mathbf{g}_k = \mathbf{D}_{k,m} \mathbf{f}_m + \boldsymbol{\eta}_k, \quad (2.12)$$

onde $\mathbf{D}_{k,m} = \mathbf{S}_k \mathbf{H}_k \mathbf{A}_{k,m}$ é uma matriz $N_1 N_2 \times M_1 M_2$.

Até agora foi apresentada a relação que uma imagem específica, k , da seqüência de imagens degradadas, teria com uma segunda imagem m , da seqüência de imagens de alta resolução. Agora será visto, através da Figura 2.3, como todas as imagens da seqüência se relacionam.

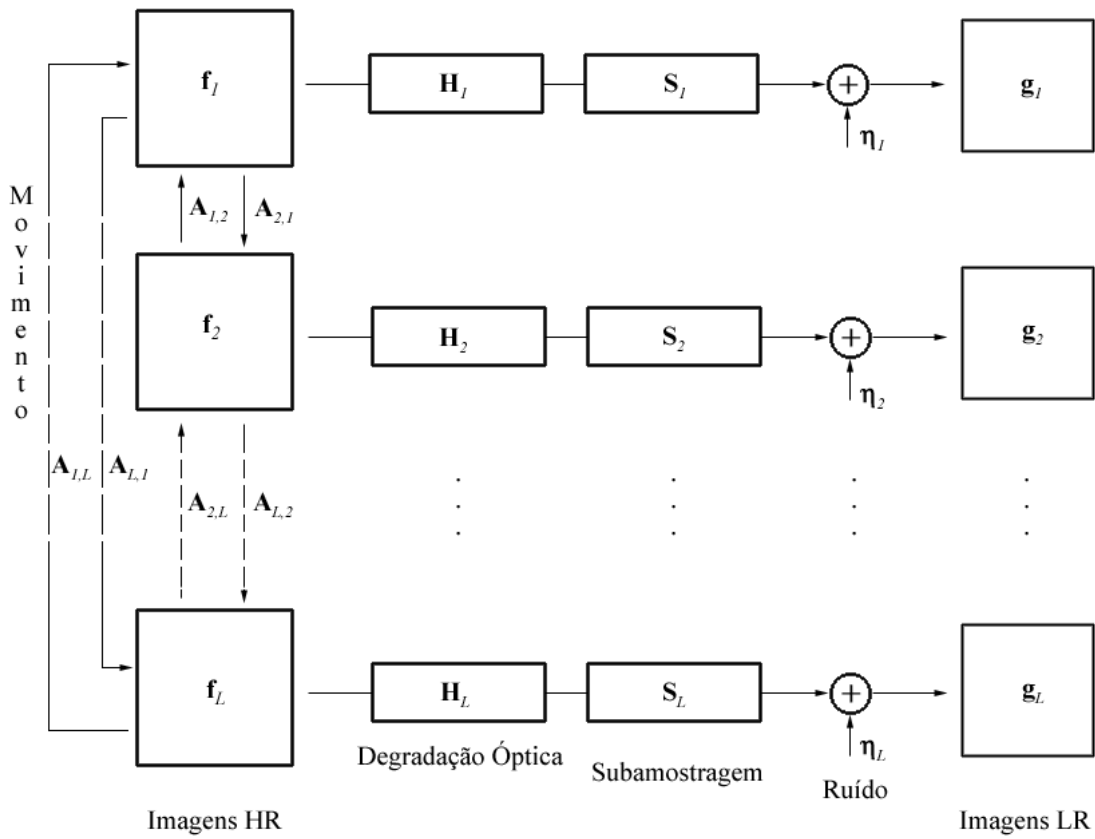


Figura 2.3: Degradação aplicada a cada uma das L imagens e o movimento entre elas.

Através da Figura 2.3 podem-se observar e entender melhor as equações apresentadas. Na seção seguinte detalha-se cada componente do processo de degradação.

2.4. Componentes de Degradação

Fica clara, na Equação (2.11), a consideração de que todos os componentes da degradação podem ser modelados como operações lineares. Agora, cada componente será observado mais detalhadamente.

2.4.1. Movimento entre as imagens

O movimento entre as imagens é modelado como o operador linear $\mathbf{A}_{k,m}$, que exprime o movimento que leva a imagem de alta resolução m a tornar-se a imagem de alta resolução k . Este movimento pode ser conhecido ou deverá ser estimado. De qualquer forma, se o movimento entre as imagens puder ser descrito, segue algum modelo definido, o que facilita a construção do operador $\mathbf{A}_{k,m}$. Os modelos de movimento são vistos no Capítulo 3. No caso da necessidade de estimação do movimento, opta-se pelo modelo de movimento que mais se aproxima do movimento real. As técnicas de estimação de movimento são encontradas em [1,13]. No Capítulo 6, os problemas relacionados à estimação são analisados.

2.4.2. Distorção óptica

A distorção óptica, \mathbf{H}_k , é um conjunto de degradações que ocorrem em parte no sistema óptico e em parte no sistema de sensores eletrônicos (como o CCD - *Charge Couple Device*). No sistema óptico, tem-se a distorção isotrópica devido à falta de foco. No conjunto eletrônico, ocorre a mistura entre os *pixels* adjacentes no sensor de imagem e distorção por movimento (*motion blur*) devido ao tempo de exposição necessário para a captura da imagem [1-3].

Essas degradações precisam ser conhecidas, ao menos as mais significativas dentro do alcance da aplicação. Caso não o sejam, podem ser empregadas técnicas de estimação da degradação óptica, como as utilizadas em procedimentos de deconvolução cega (*blind deconvolution*), mas que não vão ser abordadas neste trabalho. Uma outra forma seria estudar o dispositivo de aquisição para extrair, ou aproximar, os parâmetros desses componentes da degradação.

Embora seja possível representar degradações espaço-variantes, como anomalias em regiões específicas do sistema óptico, este trabalho considera a matriz \mathbf{H}_k como espaço-invariante, assumindo degradação óptica uniforme em toda a imagem.

2.4.3. Subamostragem

A subamostragem acontece no sensor de imagem (CCD), sendo representada por um padrão de seleção das amostras. As amostras selecionadas formam uma grade de baixa resolução, com menos amostras que a grade maior, de alta resolução. O padrão de seleção pode ser retangular, hexagonal, ou outros não retangulares. É válido lembrar que a forma da grade de subamostragem e a forma da distorção óptica de mistura entre *pixels* adjacentes estão ligadas ao tipo construção do sensor [1,2]. Neste trabalho, somente a grade retangular será utilizada em simulações.

O motivo pelo qual a matriz \mathbf{S}_k é representada em separado é apenas para realçar o processo de subamostragem. Portanto, é previsível que a matriz de subamostragem seja composta por uns e

zeros e que seja uma matriz retangular. A diferença entre os tamanhos de entrada, M_1M_2 , e de saída, N_1N_2 , demonstra a mudança na taxa de amostragem.

2.4.4. Ruído

Neste trabalho, o ruído, η_k , que é modelado por um processo aleatório com distribuição gaussiana, branco, de média zero e independente da imagem. Ele modela a adição de ruído no fotodetector. Estudos dos sensores eletrônicos de imagem demonstram que a agitação aleatória dos elétrons na presença da luz causa um ruído natural no sinal de imagem formado [2], e essa aproximação modela relativamente bem o efeito. A variância do ruído é, normalmente, indicada pelo fabricante do sensor utilizado no dispositivo de aquisição. Caso contrário, a variância pode ser determinada por métodos de estimação, ou aproximada.

Maiores informações e detalhes sobre os componentes da degradação podem ser encontrados em [1,2].

2.5. Exemplo Prático

Para que fique mais claro como o modelo de degradação é utilizado e seus componentes criados, apresenta-se um exemplo simples, no qual se observa como cada operador afeta as características do sistema.

Assume-se que o modelo de degradação tem as seguintes características:

- o movimento entre as imagens é apenas translacional global, sem a consideração do efeito das bordas perdidas e introduzidas devido ao movimento;
- o conjunto óptico, formado pelas lentes, perturbações atmosféricas e outras degradações, aplica uma filtragem ideal para uma frequência espacial F . Portanto, é o limite máximo de resolução que se poderá atingir;
- em vez de um CCD com taxa espacial de amostragem de duas vezes F (que é a taxa de Nyquist), será utilizada uma taxa duas vezes menor;
- o CCD tem grade retangular, com células fotodetectoras retangulares conforme a Figura 2.4;
- o ruído adicionado no sensor tem 5 dB (A imagem possui aproximadamente 45 dB);
- o sistema gerou 4 imagens de baixa resolução, todas com diferentes movimentos de *subpixel*, com distanciamento de $\frac{1}{2}$ (isso significa movimento de 1 *pixel* na grade de alta resolução).

Esse modelo é simples e permite usar análises no domínio da frequência. Baseando-se nas características do sistema, é possível criar as matrizes de movimento e degradação. O objetivo pode ser estimar todos os L quadros de alta resolução ou apenas um quadro. Aqui será demonstrado

2.5.2. Matriz de distorção óptica (\mathbf{H}_k) e subamostragem (\mathbf{S}_k)

Neste exemplo apenas a degradação causada no CCD será invertida. Para analisar a distorção óptica, deve-se observar a Figura 2.4.

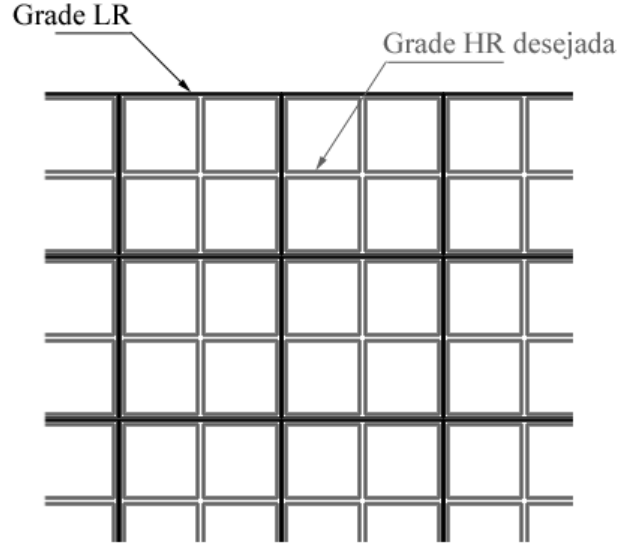


Figura 2.4: Grade representativa de subamostragem do CCD utilizado.

Observa-se que o CCD utiliza grade de amostragem com células retangulares, significando que a intensidade do *pixel* é formada pela luz que atinge esta área retangular. Esse fenômeno pode ser representado pela integral da quantidade de luz que atinge esta área. Como a degradação utilizada está no modelo discreto-discreto, é preciso relacionar os pontos da grade de alta resolução desejada com a grade de baixa resolução. Esta relação envolve a distorção óptica mais a subamostragem.

A degradação óptica pode ser modelada como a média dos *pixels* da grade de alta resolução que ficam dentro da área retangular de mistura de *pixels* adjacentes no fotosensor. Por isso, esta operação é representada como um filtro convolucional de média móvel (MA – *Moving Average*), de máscara retangular. A criação da matriz \mathbf{H}_k é muito parecida com a criação da matriz de movimento:

$$\tilde{f}_l[m_p, m_2] = f_l[m_p, m_2] * h_l[m_p, m_2] \quad (2.14)$$

8.5	7.5	8.5	9.5
4.5	3.5	4.5	5.5
8.5	7.5	8.5	9.5
12.5	11.5	12.5	13.5

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

.25	.25	0	0
.25	.25	0	0
0	0	0	0
0	0	0	0

Assim, como a operação é linear, é possível representá-la na forma matricial, e como a operação é espaço-invariante, a matriz \mathbf{H}_k será bloco-circulante, como no exemplo seguinte.

$$\begin{matrix} \tilde{f}_I \\ \begin{bmatrix} 8.5 \\ 7.5 \\ 8.5 \\ 9.5 \\ 4.5 \\ 3.5 \\ 4.5 \\ 5.5 \\ 8.5 \\ 7.5 \\ 8.5 \\ 9.5 \\ 12.5 \\ 11.5 \\ 12.5 \\ 13.5 \end{bmatrix} \end{matrix} = \begin{matrix} H_I \\ \begin{bmatrix} .25 & 0 & 0 & .25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .25 & 0 & 0 & .25 \\ .25 & .25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .25 & .25 & 0 & 0 \\ 0 & .25 & .25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .25 & .25 & 0 \\ 0 & 0 & .25 & .25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .25 & .25 \\ .25 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ .25 & .25 & 0 & 0 & .25 & .25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & .25 & .25 & 0 & 0 & .25 & .25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & .25 & .25 & 0 & 0 & .25 & .25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & .25 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & .25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & .25 & 0 & 0 & .25 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & .25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & .25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .25 & .25 & 0 & 0 & .25 & .25 \end{bmatrix} \end{matrix} \cdot \begin{matrix} f_I \\ \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 13 \\ 14 \\ 15 \\ 16 \end{bmatrix} \end{matrix}$$

A matriz de subamostragem S_k , conforme já comentado, é criada como uma matriz com zeros e uns na diagonal principal e zeros nas demais posições. As posições contendo uns correspondem aos pontos que não são dizimados; os zeros na diagonal principal indicam pontos que foram dizimados. Para realizar a redução de tamanho do sinal, as linhas compostas somente por zeros são retiradas, de forma que S_k tenha tamanho $N_1 N_2 \times M_1 M_2$, como no exemplo abaixo.

$$\begin{matrix} \tilde{f}_I \downarrow \Delta x \downarrow \Delta y \\ \begin{bmatrix} 3.5 \\ 5.5 \\ 11.5 \\ 13.5 \end{bmatrix} \end{matrix} = \begin{matrix} S_I \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{matrix} \cdot \begin{matrix} \tilde{f}_I \\ \begin{bmatrix} 8.5 \\ 7.5 \\ 8.5 \\ 9.5 \\ 4.5 \\ 3.5 \\ 4.5 \\ 5.5 \\ 8.5 \\ 7.5 \\ 8.5 \\ 9.5 \\ 12.5 \\ 11.5 \\ 12.5 \\ 13.5 \end{bmatrix} \end{matrix}$$

Agora falta apenas o ruído, mas não é necessário demonstrar a criação do vetor e a operação, dado que são triviais.

Nesse exemplo de degradação no sensor de imagem e os exemplos apresentados de construção das matrizes servem para dar uma melhor clareza sobre como os componentes físicos da degradação podem ser modelados matematicamente. Este é o primeiro passo para a solução do problema e serve, principalmente, para saber se é possível ir em frente e restaurar a resolução das imagens capturadas pelo dispositivo de aquisição.

Conclusões do Capítulo 2

Neste capítulo, apresentou-se o modelo de degradação, que é bastante genérico e pode representar bem a maioria dos efeitos reais envolvidos no processo de aquisição de imagem em dispositivos ópticos, como câmeras fotográficas digitais, filmadoras digitais, etc. A representação na notação vetorial é muito importante, pois é uma forma simples de descrever matematicamente o processo. As informações deste capítulo servirão de base para as soluções que serão apresentadas nos capítulos subsequentes.

Capítulo 3: Modelos de Movimento

Neste capítulo são estudados os modelos de movimento usados para descrever o movimento entre as imagens da seqüência. A escolha do modelo de movimento é de fundamental importância, dado que é a precisão com que o modelo de movimento escolhido aproxima o movimento real que define a viabilidade do uso das imagens da seqüência para a restauração da resolução.

Nas seções seguintes, explicam-se como o movimento real dos objetos no espaço tridimensional é convertido para o movimento aparente entre imagens e como se descreve este movimento por meio de um modelo matemático. As limitações do movimento aparente na representação do movimento real são mencionadas.

3.1. Movimento em Imagens

Assume-se que os objetos num espaço tridimensional (3D) são projetados em um plano bidimensional (2D) m , em uma condição espaço-temporal, e em um outro plano bidimensional k , numa outra condição espaço-temporal. Essas condições podem ser: um instante de tempo diferente (seqüência de vídeo), ponto de observação (câmera em posição diferente), ou ambos. Dessa forma, todo e qualquer movimento dos objetos tridimensionais e da câmera no espaço 3D, entre essas duas condições espaço-temporais, cria um movimento aparente 2D entre os planos m e k . Agora, assume-se que o movimento aparente de determinado objeto infinitesimal presente nos planos m e k possa ser descrito como:

$$\begin{aligned} \mathbf{v}_{m,k}(\mathbf{x}_m) &= \mathbf{x}_k - \mathbf{x}_m \\ \mathbf{v}_{k,m}(\mathbf{x}_k) &= \mathbf{x}_m - \mathbf{x}_k \end{aligned} \quad (3.1)$$

onde \mathbf{x}_k são as coordenadas 2D do objeto infinitesimal no plano k e \mathbf{x}_m são as coordenadas do objeto no plano m . O vetor $\mathbf{v}_{m,k}(\mathbf{x}_m)$, que descreve o movimento aparente deste objeto do plano m para o plano k , é chamado de vetor de movimento ou de vetor de deslocamento (*displacement vector*). O vetor $\mathbf{v}_{k,m}(\mathbf{x}_k)$ descreve o movimento inverso do objeto, ou seja, do plano k para o plano m . Os vetores são relacionados por $\mathbf{v}_{m,k}(\mathbf{x}_m) = -\mathbf{v}_{k,m}(\mathbf{x}_k)$.

Se forem adicionadas nas projeções k e m as considerações feitas no Capítulo 2 de amostragem suficiente e, ainda, as considerações de que não ocorre variação na iluminação, pode-se dizer que os objetos infinitesimais tornam-se *pixels*, ou têm o tamanho mínimo de um *pixel*. Então, o plano torna-se a imagem amostrada e a relação em (3.2) pode ser apresentada:

$$\begin{aligned} f_m[\mathbf{x}_m] &= f_k[\mathbf{x}_m + \mathbf{v}_{m,k}[\mathbf{x}_m]] \\ f_k[\mathbf{x}_k] &= f_m[\mathbf{x}_k + \mathbf{v}_{k,m}[\mathbf{x}_k]] \end{aligned} \quad (3.2)$$

A Equação (3.2) representa o movimento entre a imagem m e a imagem k , onde $f_k[\mathbf{x}_k]$ é o *pixel* da imagem k em coordenadas discretas globais $[m_1, m_2]$; $f_m[\mathbf{x}_m]$ é o *pixel* da imagem m , e $\mathbf{v}_{m,k}[\mathbf{x}_m]$ é o vetor que descreve o movimento que leva o *pixel* da posição \mathbf{x}_m na imagem m para a posição $\mathbf{x}_k = \mathbf{x}_m + \mathbf{v}_{m,k}[\mathbf{x}_m]$ na imagem k . O vetor $\mathbf{v}_{k,m}[\mathbf{x}_k]$ descreve o movimento inverso, que leva o *pixel* \mathbf{x}_k da imagem k para a posição $\mathbf{x}_m = \mathbf{x}_k + \mathbf{v}_{k,m}[\mathbf{x}_k]$ na imagem m . A Figura 3.1 ilustra a situação.

A Equação (3.2) deixa implícito um outro conceito, o de *resolução de movimento*. Desde que as imagens f_k e f_m são discretas e a relação apresentada leva a pontos discretos, está se assumindo que os vetores de movimento têm componentes discretas. Isto implica movimentos de unidades inteiras de *pixel*. Logo, tem-se que a melhor resolução de movimento para este caso é um *pixel*. Deve-se deixar claro aqui que, em geral, a resolução de movimento não depende da resolução da imagem. Nas situações em que a resolução do movimento é maior que a resolução da imagem, o movimento é chamado de *movimento de subpixel*.

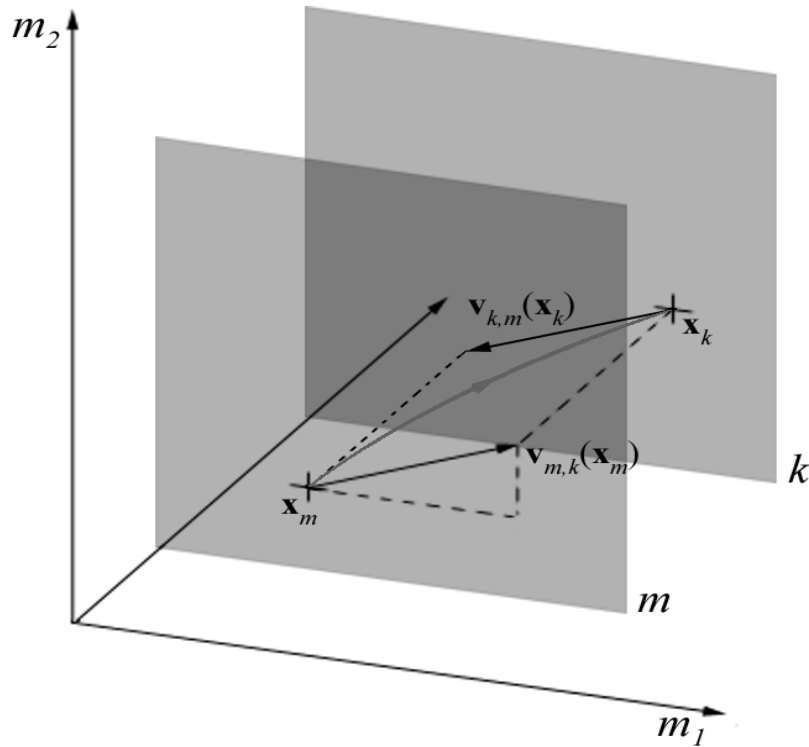


Figura 3.1: Trajetória de movimento do *pixel*.

Então, pode-se descrever a imagem f_k dados a imagem f_m e o campo de vetores $\mathbf{v}_{k,m}$; pode-se descrever também a imagem f_m dados a imagem f_k e o campo de vetores $\mathbf{v}_{m,k}$, através da relação em (3.2). Esta mesma operação é chamada de “compensação de movimento” e pode ser expressa na forma de matriz-vetor por meio de uma operação linear espaço-variante, conforme apresentado no Capítulo 2:

$$\begin{aligned} \mathbf{f}_m &= \mathbf{A}_{m,k} \mathbf{f}_k = \mathbf{A}_{k,m}^T \mathbf{f}_k, & \text{onde } \mathbf{A}_{m,k} \text{ é criado de } \mathbf{v}_{m,k} \\ \mathbf{f}_k &= \mathbf{A}_{k,m} \mathbf{f}_m = \mathbf{A}_{m,k}^T \mathbf{f}_m, & \text{onde } \mathbf{A}_{k,m} \text{ é criado de } \mathbf{v}_{k,m} \end{aligned} \quad (3.3)$$

Dessa forma, fica claro que a matriz $\mathbf{A}_{m,k}$ de transformação de movimento tem origem no campo de vetores de movimento $\mathbf{v}_{m,k}$. Uma análise mais detalhada mostra que a matriz transposta leva ao movimento inverso [10].

Os campos de vetores são limitados para aproximar o movimento aparente nos planos 2D. Apenas alguns tipos movimentos de objetos e câmeras no espaço 3D podem ser representados. Sabe-se que, no espaço 3D, os movimentos podem vir a se tornar tão complexos e misturados, com variações de luzes, reflexos e transparências de objetos, que os planos k e m não demonstram mais um movimento aparente, ou até deixam de ter alguma relação visual. Isso limita o campo de vetores a representar situações em que os movimentos 3D são relativamente simples.

Na seção seguinte, apresenta-se o modo como os modelos de movimento podem representar o movimento entre as imagens. Informações mais detalhadas podem ser encontradas em [1,3,13,22].

3.2. Composição do Modelo de Movimento

Dados a complexidade do movimento no espaço 3D e a complexidade relativa do movimento aparente 2D entre os planos k e m , pode-se representar o movimento aparente de uma forma eficaz determinando duas características: a *região* em que o movimento aparente ocorre e o *tipo de movimento* que é realizado nessa região.

A forma como a região e o tipo de movimento são determinados depende de conhecimentos *a priori* do possível movimento 3D. Por exemplo, se o movimento 3D for da câmera, pode-se dizer que toda a imagem sofrerá o mesmo movimento; logo a região é a imagem inteira. Se a câmera está se movendo de forma perpendicular à linha de projeção, então é possível dizer que o tipo de movimento da região será translacional.

Essas considerações são importantes para que o modelo represente, de forma mais precisa e mais simples possível, o movimento real. Sabe-se que a fidelidade do movimento influi diretamente na recuperação da degradação[14].

Nas seções seguintes, apresenta-se como são comumente determinadas as regiões e quais os tipos de movimentos mais utilizados.

3.2.1. Região de movimento (R)

A região de movimento, determinada por uma matriz diagonal \mathbf{R} , de tamanho $M_1M_2 \times M_1M_2$, isola um conjunto de *pixels* da imagem discreta. Esses *pixels* são selecionados em conjuntos, formando regiões que sofrem o mesmo movimento.

O exemplo mais comum de região de movimento ocorre em movimentos relativos da câmera onde os objetos são estáticos, ou seja, em movimentos que englobam a imagem toda. Portanto, a

região de movimento é a imagem toda. Tais movimentos são caracterizados como *movimentos globais*, onde a região \mathbf{R} é expressa como:

$$\mathbf{R} = \mathbf{I}, \quad (3.4)$$

onde \mathbf{I} é a matriz identidade. Exemplos desse tipo de movimento são imagens adquiridas com uma mesma câmera ou com câmeras diferentes posicionadas em diferentes lugares na aquisição da imagem \mathbf{f}_k e \mathbf{f}_m .

Movimentos aparentes de objetos relativamente grandes (maiores que um *pixel*) produzem nos planos k e m o movimento das regiões referentes a esses objetos, produzindo *movimentos regionais ou locais*. Estas regiões podem ser aproximadas a formas geométricas conhecidas, como regiões retangulares (blocos), ou podem ser determinadas por técnicas de segmentação [13]. Têm-se, assim, N regiões necessárias para representar o movimento entre as imagens, criando N matrizes que isolam os *pixels* na imagem em conjuntos diferentes, segundo:

$$\mathbf{R}_1 + \dots + \mathbf{R}_N = \mathbf{I} \quad \text{e} \quad \mathbf{R}_1 \mathbf{R}_2 \dots \mathbf{R}_N = 0, \quad (3.5)$$

onde \mathbf{R}_x , para $x=1, \dots, N$, são matrizes diagonais que isolam cada região da imagem que possui um movimento diferente.

Movimentos de pequenos objetos (do tamanho de um *pixel*), ou movimentos compostos cujas regiões não se consegue segmentar ou aproximar, produzem movimentos aparentemente individuais a cada *pixel*. Estes movimentos são chamados *movimentos de pontos individuais*. As regiões de pontos individuais atendem às especificações:

$$\sum_{x=1}^{M_1 M_2} \mathbf{R}_x = \mathbf{I} \quad \text{e} \quad \prod_{x=1}^{M_1 M_2} \mathbf{R}_x = 0 \quad (3.6)$$

Pode-se representar a matriz de movimento separando o movimento de cada região, segundo:

$$\mathbf{A}_{k,m} = \mathbf{R}_1 \mathbf{A}_{k,m}^{\mathbf{R}_1} + \dots + \mathbf{R}_N \mathbf{A}_{k,m}^{\mathbf{R}_N} \quad (3.7)$$

Cada matriz $\mathbf{A}_{k,m}^{\mathbf{R}_x}$ determina o tipo de movimento sofrido por cada região \mathbf{R}_x , onde $x=1, \dots, N$. A Figura 3.2 demonstra os exemplos de regiões de movimento apresentados.

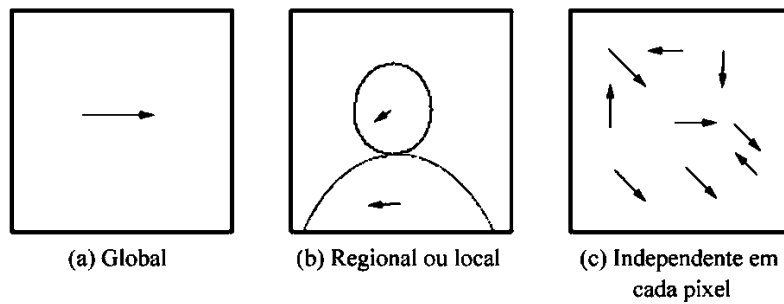


Figura 3.2: Regiões de movimento.

3.2.2. Tipos de movimento

O tipo de movimento aparente observado em uma região de movimento do plano 2D, entre as imagens f_k e f_m , é derivado das características 3D da cena. Essas características são, por exemplo, o modelo de movimento 3D de um objeto ou câmera, a função de superfície 3D do objeto ou cena e a geometria de projeção da câmera. A Tabela 3.1, extraída de [13], demonstra vários tipos de movimento, o modo como o campo de vetores é gerado a partir dos parâmetros do tipo movimento e de onde os movimentos são derivados. A Figura 3.3 exemplifica os três primeiros movimentos da Tabela 3.1, onde $\mathbf{x}=(x,y)$.

Tabela 3.1: Tipos de movimento.

Movimento Aparente 2D			Modelo 3D		Modelo de Câmera
Nome	Número de parâmetros	Campo de Vetores	Superfície 3D	Movimento 3D	
Translacional	2	$\mathbf{v}(\mathbf{x}) = [a_0 \ b_0]^T$	Arbitrária	Translação rígida 3D	Ortográfica
<i>Affine</i>	6	$\mathbf{v}(\mathbf{x}) = \begin{bmatrix} a_{10} & a_{01} \\ b_{10} & b_{01} \end{bmatrix} \mathbf{x} + \begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$	Planar	3D <i>Affine</i>	Ortográfica
Projetivo Linear	8	$\mathbf{v}(\mathbf{x}) = \begin{bmatrix} \frac{a_1 + a_2 x + a_3 y}{1 + a_4 x + b_4 y} \\ \frac{b_1 + b_2 x + b_3 y}{1 + a_4 x + b_4 y} \end{bmatrix} - \mathbf{x}$	Planar	3D <i>Affine</i>	Perspectiva
Quadrático	12	$\mathbf{v}(\mathbf{x}) = \begin{bmatrix} a_0 + a_{10}x + a_{01}y + a_{20}x^2 + a_{11}xy + a_{02}y^2 \\ b_0 + b_{10}x + b_{01}y + b_{20}x^2 + b_{11}xy + b_{02}y^2 \end{bmatrix}$	Parabólico	3D <i>Affine</i>	Ortográfica
Amostrado	2 por Δ^2 pixels	$\mathbf{v}(\mathbf{x}) = \sum_{i,j} \begin{bmatrix} a_{ij} \\ b_{ij} \end{bmatrix} H(x - \Delta i, y - \Delta j)$	“Suave” como especificado pelo núcleo de interpolação H		Arbitrário
Polinomial	2KI Adaptativo	$\mathbf{v}(\mathbf{x}) = \sum_{i,j \in K} \begin{bmatrix} a_{ij} \\ b_{ij} \end{bmatrix} [x^i \ y^j]$	“Suave” como especificado por K		Arbitrário

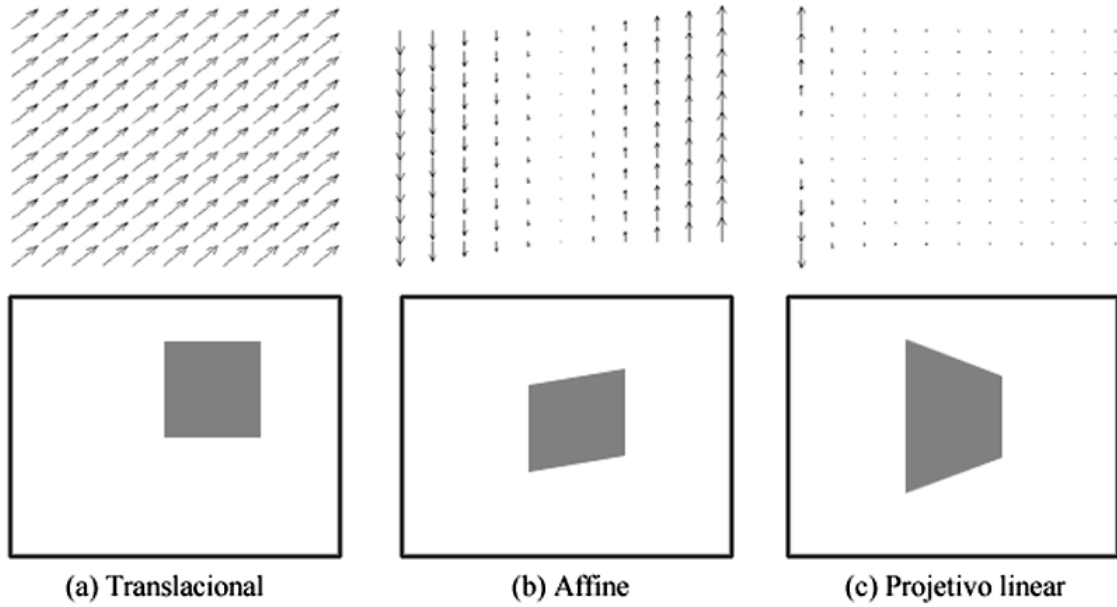


Figura 3.3: Exemplos de tipos de movimento.

Em aplicações nas quais o movimento real é controlado ou conhecido, pode-se chegar com exatidão ao movimento aparente a partir das características 3D da cena e, conseqüentemente, ao campo de vetores. Porém, na maioria das aplicações, as condições das quais o movimento aparente é derivado são vagamente conhecidas. Isso permite apenas escolher um tipo de movimento, aquele que mais aproxima o movimento observado, e estimar seus parâmetros.

Também é importante notar uma restrição nos movimentos. De forma alguma os movimentos 3D existentes na cena devem causar na projeção 2D discreta o efeito de *aliasing*, pois isso invalida as equações em (3.1), (3.2) e (3.3). Por exemplo, o movimento espacial de afastamento de um objeto 3D qualquer de uma câmera que utiliza projeção de imagem na forma de perspectiva. Observa-se que, na condição espaço-temporal do plano m , o objeto 3D está próximo da câmera, formando uma imagem f_m (projeção discreta) cujo objeto 3D projetado aparece grande. Já, na condição espaço-temporal do plano k , o objeto 3D está afastado da câmera, formando, nesse instante, uma imagem f_k em que o objeto 3D projetado aparenta ser menor. Isso significa que as frequências espaciais aumentaram da imagem f_m para a imagem f_k por causa do movimento do objeto 3D de m para k . Esse movimento é aceitável até o ponto em que não há *aliasing* na formação da imagem f_k , ou seja, a taxa de amostragem espacial da imagem é suficiente, pois está acima da taxa de Nyquist nas duas imagens. Caso contrário, é necessário uma taxa de amostragem espacial maior.

Este exemplo é ilustrado pela Figura 3.4. Observa-se que, com o afastamento do objeto retangular e com a baixa taxa de amostragem, as linhas diagonais que compõem sua textura sofrem *aliasing*. Isso é notado pela aparente mudança na direção das linhas diagonais.

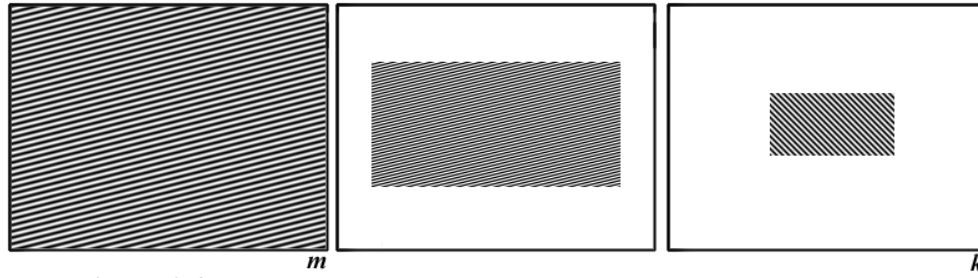


Figura 3.4: *Aliasing* na formação de uma das imagens com movimento.

Respeitando todas as regras mencionadas neste capítulo, pode-se achar um modelo de movimento que aproxima o movimento aparente das imagens da sequência. Através dele chega-se ao campo de vetores e , então, às matrizes $A_{k,m}$, que são usadas na formulação da degradação, completando, assim, toda formulação matemática da degradação apresentada no Capítulo 2. Se não houver um modelo que possa ser usado, por não representar adequadamente o movimento entre a imagem f_m e as outras imagens da sequência, provavelmente não será possível fazer uso dessas outras imagens na recuperação da degradação.

3.3. Modelo de Movimento com *Inliers* e *Outliers*

Como foi visto até agora, o movimento entre as imagens é composto por diferentes regiões da imagem que sofrem diferentes tipos de movimento. Também foi visto que, em muitas situações, o movimento aparente pode se tornar tão complexo que não se podem relacionar as imagens através de movimento. Entretanto, existem várias situações em que parte das imagens ainda pode ser relacionada. Portanto, assim como se podem considerar regiões com diferentes tipos de movimento, também é possível incluir no modelo regiões onde as imagens não conseguem ser relacionadas pelo movimento, possibilitando que ao menos uma parte da imagem seja usada no processo de restauração da resolução.

Têm-se, então, dois novos tipos de região: para as regiões onde o movimento não pode ser relacionado, dá-se o nome de *outliers*; para regiões onde o movimento pode ser relacionado, como no modelo visto anteriormente, dá-se o nome de *inliers*. Diz-se, então, que o modelo comporta a presença de *outliers*, cuja existência no movimento entre as imagens discretas é mais comum do que parece. Observem-se alguns exemplos:

- objetos (regiões) que sofrem movimentos muito complexos que não produzem um movimento aparente que possa ser representado pelos vetores de movimento;
- partes de objetos que entram ou saem das bordas da imagem;
- regiões cobertas ou expostas pelo movimento de um outro objeto.

Logo, a presença de *outliers* implica a mudança do equacionamento que relaciona as duas imagens.

3.3.1. Mudança no modelo de movimento devido aos *inliers* e *outliers*

Em virtude da presença de *outliers* não se pode descrever completamente uma imagem \mathbf{f}_k baseando-se apenas nos vetores de movimento $\mathbf{v}_{k,m}$, ou na transformação de movimento $\mathbf{A}_{k,m}$ e na imagem \mathbf{f}_m . Isso faz com que a relação em (3.3) seja inválida sob a presença de *outliers*. Assim, apenas os *pixels inliers* podem ser relacionados, segundo:

$$\begin{aligned}\mathbf{f}_m^I &= \mathbf{A}_{m,k} \mathbf{f}_k^I \\ \mathbf{f}_k^I &= \mathbf{A}_{k,m} \mathbf{f}_m^I\end{aligned}\quad (3.8)$$

onde \mathbf{f}_m^I é um vetor de tamanho $M_1 M_2 \times 1$. Nos *pixels inliers* tem-se que $f_m^I[\mathbf{x}] = f_m[\mathbf{x}]$; já, nos *pixels* considerados *outliers*, $f_m^I[\mathbf{x}] = 0$. O mesmo é válido para \mathbf{f}_k^I . A multiplicação das imagens pelas devidas máscaras, segundo (3.9), pode isolar os *inliers* e os *outliers* nas imagens.

$$\mathbf{f}^I = \mathbf{i} \bullet \mathbf{f} = \begin{bmatrix} i_1 \\ \vdots \\ i_n \end{bmatrix} \bullet \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} = \begin{bmatrix} i_1 f_1 \\ \vdots \\ i_n f_n \end{bmatrix} \quad (3.9)$$

Na Equação (3.9), a operação “ \bullet ” denota multiplicação de elemento por elemento entre os vetores \mathbf{i} , que representa a máscara de *inliers*, e \mathbf{f} , que representa a imagem. Aplica-se essa operação tanto na imagem \mathbf{f}_k quanto em \mathbf{f}_m , levando às seguintes relações:

$$\begin{aligned}\mathbf{f}_m^I &= \mathbf{i}_m \bullet \mathbf{f}_m = \mathbf{i}_m \bullet (\mathbf{A}_{m,k} \mathbf{f}_k) = \mathbf{A}_{m,k} (\mathbf{i}_k \bullet \mathbf{f}_k) \\ \mathbf{f}_k^I &= \mathbf{i}_k \bullet \mathbf{f}_k = \mathbf{i}_k \bullet (\mathbf{A}_{k,m} \mathbf{f}_m) = \mathbf{A}_{k,m} (\mathbf{i}_m \bullet \mathbf{f}_m)\end{aligned}\quad (3.10)$$

onde os vetores \mathbf{i}_k e \mathbf{i}_m são vetores de tamanho $M_1 M_2 \times 1$, compostos por uns e zeros, uns nas posições referentes aos *pixels inliers* e zeros nas posições referentes aos *pixels outliers*.

Essas mesmas operações podem ser realizadas com matrizes dentro das regras da álgebra linear, como:

$$\begin{aligned}\mathbf{f}_m^I &= \mathbf{I}_m \mathbf{f}_m = \mathbf{I}_m \mathbf{A}_{m,k} \mathbf{f}_k = \mathbf{A}_{m,k} \mathbf{I}_k \mathbf{f}_k \\ \mathbf{f}_k^I &= \mathbf{I}_k \mathbf{f}_k = \mathbf{I}_k \mathbf{A}_{k,m} \mathbf{f}_m = \mathbf{A}_{k,m} \mathbf{I}_m \mathbf{f}_m\end{aligned}\quad (3.11)$$

onde as matrizes \mathbf{I}_m e \mathbf{I}_k são matrizes diagonais, de tamanho $M_1 M_2 \times M_1 M_2$, cujos elementos da diagonal principal são os mesmos dos vetores \mathbf{i}_m e \mathbf{i}_k , respectivamente. Para evitar confusão com a notação \mathbf{I} , da matriz identidade, as matrizes de *inliers* estarão sempre acompanhadas do subscrito k , ou m , referente à imagem em questão. Isso leva a uma interessante mudança na transformação de movimento $\mathbf{A}_{k,m}$:

$$\begin{aligned}\mathbf{A}_{m,k}^I &= \mathbf{I}_m \mathbf{A}_{m,k} = \mathbf{A}_{m,k} \mathbf{I}_k \\ \mathbf{A}_{k,m}^I &= \mathbf{I}_k \mathbf{A}_{k,m} = \mathbf{A}_{k,m} \mathbf{I}_m\end{aligned}\quad (3.12)$$

onde a matriz $\mathbf{A}_{k,m}^I$ expressa a transformação de movimento dos *inliers*. Neste caso, não existe uma matriz $\mathbf{A}_{k,m}$ que expresse a transformação de movimento de todos os *pixels*. O exemplo abaixo é uma modificação do exemplo da Seção 2.5.1, porém considerando as bordas como *outliers*.

$$\begin{array}{c} \mathbf{f}_2' \\ \begin{array}{|c|} \hline 0 \\ 0 \\ 0 \\ 0 \\ 4 \\ 1 \\ 2 \\ 3 \\ 8 \\ 5 \\ 6 \\ 7 \\ 12 \\ 9 \\ 10 \\ 11 \\ \hline \end{array} \end{array} = \begin{array}{c} \mathbf{A}_{2,1}^I \\ \begin{array}{|cccccccccccccccc|} \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline \end{array} \end{array} \cdot \begin{array}{c} \mathbf{f}_1 \\ \begin{array}{|c|} \hline 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 13 \\ 14 \\ 15 \\ 16 \\ \hline \end{array} \end{array}$$

Da mesma forma que a representação dos *inliers*, a representação dos *outliers* pode ser dada por:

$$\begin{aligned} \mathbf{f}_m^O &= \mathbf{o}_m \bullet \mathbf{f}_m = \mathbf{O}_m \mathbf{f}_m = \mathbf{f}_m - \mathbf{A}_{m,k}^I \mathbf{f}_k \\ \mathbf{f}_k^O &= \mathbf{o}_k \bullet \mathbf{f}_k = \mathbf{O}_k \mathbf{f}_k = \mathbf{f}_k - \mathbf{A}_{k,m}^I \mathbf{f}_m \end{aligned} \quad (3.13)$$

Os vetores \mathbf{o}_k e \mathbf{o}_m e as matrizes \mathbf{O}_k e \mathbf{O}_m seguem as mesmas regras dos vetores \mathbf{i}_k e \mathbf{i}_m e das matrizes \mathbf{I}_k e \mathbf{I}_m . Porém, agora, os elementos “uns” representam os *outliers* e os elementos “zeros” representam os *inliers*. Logo:

$$\begin{aligned} \mathbf{o}_m \bullet \mathbf{i}_m &= \mathbf{0}, \quad \mathbf{o}_m + \mathbf{i}_m = \mathbf{1}, \quad \mathbf{O}_m \mathbf{I}_m = \mathbf{O}, \quad \mathbf{O}_m + \mathbf{I}_m = \mathbf{I} \\ \mathbf{o}_k \bullet \mathbf{i}_k &= \mathbf{0}, \quad \mathbf{o}_k + \mathbf{i}_k = \mathbf{1}, \quad \mathbf{O}_k \mathbf{I}_k = \mathbf{O}, \quad \mathbf{O}_k + \mathbf{I}_k = \mathbf{I} \end{aligned} \quad (3.14)$$

onde $\mathbf{0}$ é um vetor de zeros, $\mathbf{1}$ é um vetor de uns, \mathbf{O} uma matriz de zeros e \mathbf{I} , a matriz identidade.

O uso dos vetores que representam as máscaras é interessante principalmente porque as máscaras de *inliers* devem se relacionar entre si com a mesma matriz de movimento, como apresentado em (3.15). Porém, as máscaras de *outliers* não podem ser relacionadas diretamente.

$$\begin{aligned} \mathbf{i}_m &= \mathbf{A}_{m,k} \mathbf{i}_k = \mathbf{A}_{m,k}^I \mathbf{i}_k = \mathbf{A}_{m,k}^I \mathbf{1} \\ \mathbf{i}_k &= \mathbf{A}_{k,m} \mathbf{i}_m = \mathbf{A}_{k,m}^I \mathbf{i}_m = \mathbf{A}_{k,m}^I \mathbf{1} \end{aligned} \quad (3.15)$$

A Figura 3.5 demonstra duas imagens com movimento e as respectivas máscaras de *inliers* (branco) e *outliers* (em preto) embaixo. Nas imagens os *outliers* são as bordas e parte de um objeto que entra na cena.

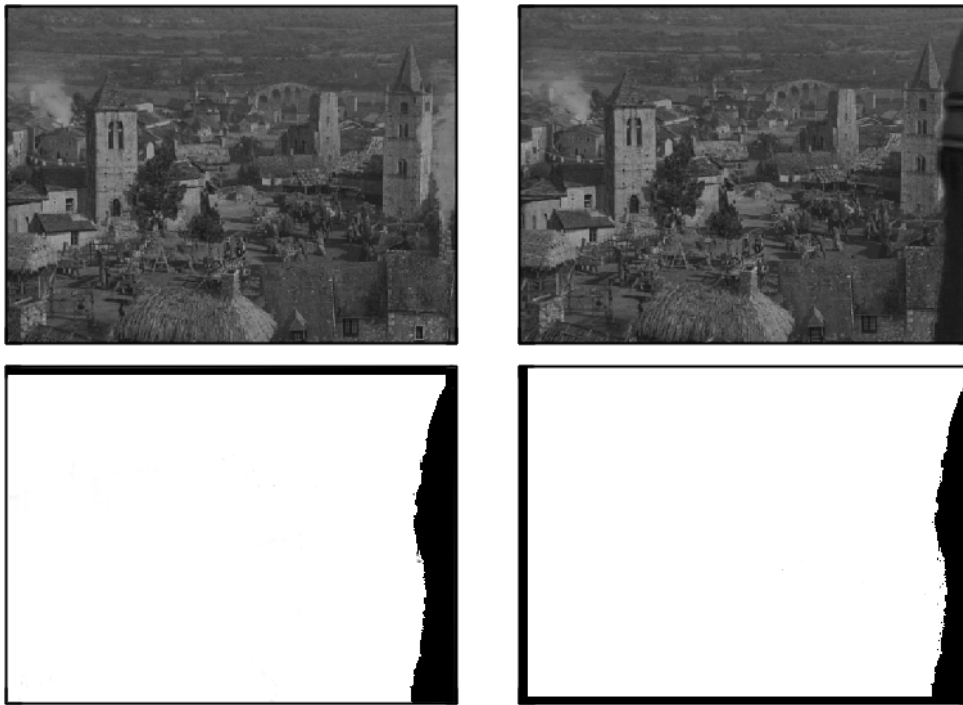


Figura 3.5: Regiões de *outliers* entre as imagens marcadas em preto.

Alguns trabalhos mencionam o problema dos *outliers* [1,13-14,22], mas não apresentam uma formulação apropriada. Esses vetores, matrizes e operações introduzidas nesta seção serão usados nos capítulos posteriores nas modificações das equações para trabalhar com seqüências de imagens com a presença de *inliers* e *outliers*.

Conclusões do Capítulo 3

Foi visto que é possível descrever o movimento aparente entre as imagens baseando-se no conhecimento *a priori* do movimento real 3D. Através deste conhecimento pode-se escolher um modelo de movimento adequado que relacione as imagens. O modelo de movimento pode ser composto de várias regiões que sofrem diferentes tipos de movimento.

A existência de *outliers* no movimento não é desejável, mas proíbe a utilização da imagem para restauração. Desde que a região de *outliers* seja reconhecida e isolada, a região útil, ou *inliers*, pode ser usada.

Relacionando as imagens pelo movimento, seja integral ou parcialmente, pode-se fazer uso da informação adicional, pertencente a elas, nas técnicas de restauração da resolução.

Capítulo 4: Soluções para Inverter a Degradação

Neste capítulo, apresenta-se o desenvolvimento da linha de solução adotada para inverter a da degradação nas imagens da seqüência. Conforme o Capítulo 2, o modelo de degradação pode representar vários tipos de movimento, diferentes degradações no fluxo óptico e vários padrões de subamostragem. O modelo de degradação requer representação com notação vetorial por causa da complexidade de suas operações. Como a degradação é modelada como um sistema linear, a correção da degradação é tratada como solução de sistemas lineares.

Nas seções seguintes, a solução para a degradação é formulada como solução de sistemas lineares. A linha de solução adotada é apresentada e explicada.

4.1. Análise da Solução de Sistemas Lineares

Primeiramente, degradação é observada como uma transformação linear, ou sistema linear, conforme abaixo:

$$\mathbf{g}_k = \mathbf{D}_{k,m} \mathbf{f}_m + \boldsymbol{\eta}_k \quad (4.1)$$

Desconsiderando, inicialmente, a existência do ruído $\boldsymbol{\eta}_k$, a solução de (4.1) poderia ser encontrada se existisse uma matriz inversa de $\mathbf{D}_{k,m}$. O problema é que o sistema é, por natureza, subdeterminado [4-5], o espaço nulo da matriz $\mathbf{D}_{k,m}$ não é vazio. Mais precisamente, o posto (*rank*) p é menor que o número de colunas, que é $M_1 M_2$. Essa condição de posto incompleto, ou $p < M_1 M_2$, do sistema linear leva à não-unicidade de soluções [1,4-5].

O motivo que leva a matriz $\mathbf{D}_{k,m}$ a ter deficiência de posto é a perda de informação sobre \mathbf{f}_m durante a aquisição de cada imagem \mathbf{g}_k . Esta perda de informação é gerada por regiões onde o movimento não pode ser relacionado através do modelo de movimento (*outliers*), os zeros nas degradações ópticas e, principalmente, os *pixels* eliminados no processo de subamostragem (este último é o maior responsável pela subdeterminação). Portanto, o sistema tem mais incógnitas que equações, mais precisamente, $N_1 N_2$ equações, das quais apenas p são linearmente independentes, com $M_1 M_2$ incógnitas ($M_1 M_2 > N_1 N_2 \geq p$).

Considerando o ruído, a Equação (4.1) pode não ter solução, mesmo quando a matriz $\mathbf{D}_{k,m}$ não tem problemas de deficiência de posto, pois o sistema pode ser inconsistente. Além disso, uma possível solução inversa poderia ter o ruído excessivamente amplificado. Em virtude dessas condições, o sistema linear apresentado pela Equação (4.1) é um sistema mal-condicionado (*ill-conditioned*) [4-5,24-25].

4.1.1. Seqüência de imagens e solução idealizada

Como as matrizes $\mathbf{D}_{k,m}$ são subdeterminadas, um novo sistema, onde as várias imagens da seqüência observada são concatenadas com a finalidade de se obter um sistema com mais equações que o Sistema (4.1), pode ser considerado.

$$\begin{bmatrix} \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_L \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{1,m} \\ \vdots \\ \mathbf{D}_{L,m} \end{bmatrix} \mathbf{f}_m + \begin{bmatrix} \boldsymbol{\eta}_1 \\ \vdots \\ \boldsymbol{\eta}_L \end{bmatrix}, \quad \text{ou} \quad \mathbf{g} = \mathbf{D}_m \mathbf{f}_m + \boldsymbol{\eta} \quad (4.2)$$

No Sistema (4.2) \mathbf{g} é um vetor de tamanho $LN_1N_2 \times 1$, composto pelos L vetores \mathbf{g}_k , de tamanho $N_1N_2 \times 1$, concatenados; \mathbf{D}_m é uma matriz $LN_1N_2 \times M_1M_2$, composta pelas L matrizes $\mathbf{D}_{k,m}$ concatenadas, e o vetor $\boldsymbol{\eta}$ também é $LN_1N_2 \times 1$.

Agora, o novo sistema linear possui um número maior de equações. Espera-se que as imagens possuam informação nova, seja por movimento *subpixel*, seja por degradações diferentes em cada quadro. Essa informação nova significa que o sistema (4.2) possui mais equações linearmente independentes que (4.1), aumentando o posto do sistema de (4.1) para (4.2).

O aumento do posto traduz-se na diminuição do tamanho do conjunto de soluções possíveis. Se, por um instante, for desconsiderado o ruído e o sistema tiver imagens suficientes, com nova informação em cada quadro, a ponto de se obter uma matriz \mathbf{D}_m não singular ($p=M_1M_2$ e $LN_1N_2=M_1M_2$), a inversa de \mathbf{D}_m pode ser encontrada e o sistema solucionado. Lembra-se que todo sistema não singular tem solução e esta é única [4-5].

Para demonstrar isso, considera-se um exemplo simples, similar ao exemplo prático da Seção 2.5. Têm-se, então, quatro imagens onde a degradação causou perda da resolução em duas vezes. O movimento é translacional global, conhecido, de $\frac{1}{2}$ de *pixel*, em diferentes direções em cada imagem. A degradação óptica é espaço-invariante e não contém zeros no eixo da freqüência. A subamostragem é em grade retangular e não há ruído, conforme a Figura 4.1.

Assim, cada imagem da seqüência usada na recuperação da imagem de alta resolução equivale ao uso de mais $(M_1M_2)/4$ equações linearmente independentes em (4.2). Logo, o sistema teria solução inversa e a imagem \mathbf{f}_m poderia ser recuperada.

Infelizmente, essa situação *idealizada* não ocorre, dado que o ruído existe e, na maioria das aplicações, não se pode controlar a nova informação adicionada. Isso porque ou não se consegue obter todas as imagens necessárias, ou há zeros no eixo da freqüência na degradação óptica. Os zeros no eixo da freqüência, para sistemas espaço-invariantes, são autovalores nulos na matriz \mathbf{H}_k , deixando a matriz \mathbf{H}_k e a matriz \mathbf{D}_m com deficiência de posto.

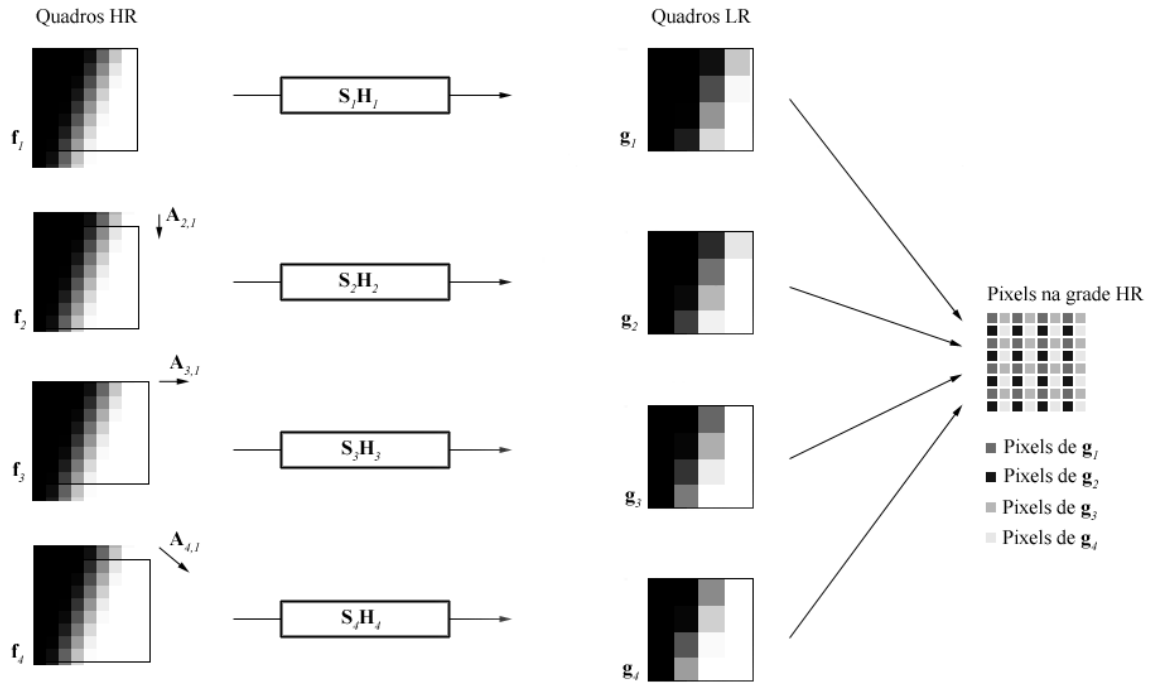


Figura 4.1: Composição dos quatro quadros LR na grade HR para reconstrução.

Sendo assim, todas as formas de solução para a degradação devem resolver os possíveis problemas do sistema linear da Equação (4.2): a inexistência de solução (devido à presença de ruído), o mal-condicionamento da matriz D_m (devido aos autovalores pequenos sob ruído) e a não-unicidade de solução (devido aos valores singulares nulos ou a subdeterminação).

4.2. Solução por Otimização e Regularização

Dadas as características dos sistemas lineares em (4.1) e (4.2), buscam-se alternativas para contornar os problemas de inexistência de solução, de mal-condicionamento e de não-unicidade de solução (se a solução existir). O conjunto otimização e regularização é comumente utilizado para resolver esses problemas. A otimização é aplicada de forma a escolher uma solução de acordo com algum critério que permita a existência de solução. Como o sistema é mal-condicionado, ou seja, uma pequena variação nos dados em g , em virtude do ruído η , causa uma grande variação na solução de f_m , ele deve ser regularizado, de forma que pequenas variações em g não prejudiquem a solução.

Em seguida, apresentam-se as soluções para a correção dos problemas de inexistência de solução, unicidade de solução e mal-condicionamento através de otimização e regularização.

4.2.1. Solução inversa generalizada

Considerando a degradação das várias imagens da sequência, de acordo com o sistema linear (4.2), o sistema pode ser reformulado de forma a se obter:

$$\boldsymbol{\eta} = \mathbf{g} - \mathbf{D}_m \mathbf{f}_m \quad (4.3)$$

Através da Equação (4.3), pode-se evidenciar que o sistema é perturbado pelo ruído. Assumido, então, que o ruído $\boldsymbol{\eta}$ é a fonte de erro nos dados do sistema, pode-se aplicar a solução de *Mínimos Quadrados* (LS – *Least Squares*) para minimizá-lo:

$$\hat{\mathbf{f}}_m^{LS} = \arg \min_{\mathbf{f}_m} \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2^2, \quad (4.4a)$$

onde \mathbf{f}_m^{LS} denota a estimativa de \mathbf{f}_m pelo método de Mínimos Quadrados (LS) e a operação $\|\cdot\|_2$ denota norma Euclidiana. Vale lembrar que o método de mínimos quadrados é determinístico, o que significa que não estão sendo consideradas as características estatísticas do ruído $\boldsymbol{\eta}$, visto aqui como o erro.

A otimização da Equação (4.4a) leva ao sistema linear apresentado em (4.4b), também conhecido por *Equações Normais*:

$$\mathbf{D}_m^T \mathbf{D}_m \mathbf{f}_m = \mathbf{D}_m^T \mathbf{g} \quad (4.4b)$$

Neste novo sistema, as equações não são mais inconsistentes e o sistema agora teria solução, a qual será $\mathbf{f}_m^{LS} = (\mathbf{D}_m^T \mathbf{D}_m)^{-1} \mathbf{D}_m^T \mathbf{g}$ se o espaço nulo da matriz \mathbf{D}_m for vazio. O índice “ T ” denota a matriz transposta em caso real e a transposta complexa conjugada no caso complexo. A solução de Mínimos Quadrados também é escolhida pelo fato de $\mathbf{D}_m^T \mathbf{D}_m$ ser uma matriz simétrica e ter seus autovalores maiores ou iguais a zeros, o que facilita a inversão.

Como existem valores singulares nulos em \mathbf{D}_m , seu espaço nulo não é vazio, $\mathbf{D}_m^T \mathbf{D}_m$ tem autovalores nulos e ainda não há uma solução única. Por isso, a solução de norma mínima é adotada, modificando as equações em (4.4a) para as equações em (4.5). A solução de norma mínima indica que os autovalores nulos em $\mathbf{D}_m^T \mathbf{D}_m$ não serão invertidos. A solução em (4.5) é chamada de *Pseudo-inversa* ou solução *Inversa Generalizada* [4]:

$$\mathbf{f}_m^+ = \arg \min_{\mathbf{f}_m} \|\mathbf{f}_m\|_2 \quad \text{restrito à} \quad \arg \min_{\mathbf{f}_m} \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2^2, \quad (4.5)$$

onde \mathbf{f}_m^+ denota a solução inversa generalizada. Esse equacionamento é, às vezes, expresso através de $\mathbf{f}_m^+ = \mathbf{D}_m^+ \mathbf{g}$, onde a matriz \mathbf{D}_m^+ é a pseudo-inversa de \mathbf{D}_m .

Nota-se que, se a matriz \mathbf{D}_m fosse bloco circulante, poderia ser diagonalizada utilizando a Transformada Bidimensional de Fourier, apresentada como a transformação linear \mathbf{F} :

$$\mathbf{D}_m = \mathbf{F}^{-1} \tilde{\mathbf{D}}_m \mathbf{F} \quad \text{e} \quad \mathbf{D}_m^T = \mathbf{F}^{-1} \tilde{\mathbf{D}}_m^* \mathbf{F}, \quad (4.6)$$

onde $\tilde{\mathbf{D}}_m$ é uma matriz diagonal cujos elementos denotam o espectro de Fourier, $\tilde{\mathbf{D}}_m^*$ é o complexo conjugado de $\tilde{\mathbf{D}}_m$. Neste caso, a resposta da inversa generalizada poderia ser:

$$(\tilde{\mathbf{D}}_m^* \tilde{\mathbf{D}}_m) \tilde{\mathbf{f}}_m^+ = \tilde{\mathbf{D}}_m^* \tilde{\mathbf{g}}, \quad (4.7)$$

onde $\tilde{\mathbf{f}}_m^+$ é um vetor onde os valores são os componentes da transformada de Fourier de \mathbf{f}_m^+ e $\tilde{\mathbf{g}}$ são os componentes de Fourier de \mathbf{g} . Isso caracteriza os seguintes resultados para cada componente:

$$\tilde{\mathbf{f}}_m^+[i] = \begin{cases} \frac{1}{\tilde{\mathbf{d}}_m[i]} \tilde{\mathbf{g}}[i], & |\tilde{\mathbf{d}}_m[i]| \neq 0 \\ 0 & \text{outros} \end{cases}, \text{ onde } i = 1, \dots, M_1 M_2 \quad (4.8)$$

Assim, a inversa generalizada realiza uma inversão nos elementos $\tilde{\mathbf{d}}_m[i]$, os componentes diagonais de $\tilde{\mathbf{D}}_m$, não nulos e zera os demais [1].

Na maioria dos casos, a matriz \mathbf{D}_m não é bloco circulante, portanto, utiliza-se a *Decomposição em Valores Singulares* (SVD – *Singular Value Decomposition*) para explicar os efeitos da pseudo-inversa. Dessa forma, sendo \mathbf{D}_m uma matriz $LN_1 N_2 \times M_1 M_2$ (será dito $n \times m$ para simplificação das próximas equações), pode-se obter a equação:

$$\mathbf{D}_m = \mathbf{U} \mathbf{S} \mathbf{V}^T = \sum_{i=1}^p \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \quad (4.9)$$

onde $\mathbf{U} = [\mathbf{u}_1 \dots \mathbf{u}_n]$ é uma matriz unitária $n \times n$, $\mathbf{V} = [\mathbf{v}_1 \dots \mathbf{v}_m]$ é uma matriz unitária $m \times m$ e \mathbf{S} é uma matriz diagonal $n \times m$ com os elementos $\sigma_1, \sigma_2, \dots, \sigma_q$ arranjados na diagonal principal, e zeros no resto, sendo $q = \min(m, n) = \min(LN_1 N_2, M_1 M_2)$. Os vetores coluna ortonormais \mathbf{u}_i de \mathbf{U} são chamados vetores singulares esquerdos e também são os autovetores de $\mathbf{D}_m \mathbf{D}_m^T$. Os vetores coluna ortonormais \mathbf{v}_i de \mathbf{V} são chamados vetores singulares direitos e também são os autovetores de $\mathbf{D}_m^T \mathbf{D}_m$. O conjunto $\{\sigma_i, \mathbf{u}_i, \mathbf{v}_i\}$, onde $1 \leq i \leq q$, é chamado sistema singular de \mathbf{D}_m . Assim, se o posto da matriz \mathbf{D}_m é p , σ_i satisfaz:

$$\sigma_1 \geq \dots \geq \sigma_p > \sigma_{p+1} = \dots = \sigma_q = 0 \quad (4.10)$$

Embora calcular numericamente a SVD para matrizes maiores que um tamanho modesto seja computacionalmente custoso, a SVD é uma ferramenta extremamente útil. Pode-se, inclusive, escrever a solução inversa generalizada em função do sistema singular de \mathbf{D}_m [1].

$$\mathbf{f}_m^+ = \sum_{i=1}^p \frac{\mathbf{u}_i^T \mathbf{g}}{\sigma_i} \mathbf{v}_i \quad (4.11)$$

Essa expressão é válida para qualquer \mathbf{D}_m . Aqui os dados observados \mathbf{g} são decompostos na nova base \mathbf{u}_i , produzindo os coeficientes $\mathbf{u}_i^T \mathbf{g}$, que são escalados pela relação $1/\sigma_i$ e então usados como peso para o retorno à base anterior \mathbf{v}_i . Nota-se que a soma ocorre até p , que são os componentes σ_i maiores que zero.

A solução inversa generalizada, embora tenha uma solução única, leva a resultados “instáveis” causados pela perturbação do ruído. Observa-se na Equação (4.12):

$$\frac{\mathbf{u}_i^T \mathbf{g}}{\sigma_i} = \mathbf{v}_i^T \mathbf{f}_m^+ + \frac{\mathbf{u}_i^T \boldsymbol{\eta}}{\sigma_i} \quad (4.12)$$

A Equação (4.12) demonstra que o ruído η poderá ser excessivamente amplificado se houver algum valor singular muito pequeno, o que normalmente ocorre com os valores singulares σ_i próximos aos zeros. Isso demonstra a necessidade da regularização quando o sistema é mal-condicionado.

O gráfico da Figura 4.2 demonstra os valores singulares de uma matriz e as perturbações na inversão generalizada causadas pelo ruído em meio a valores singulares pequenos.

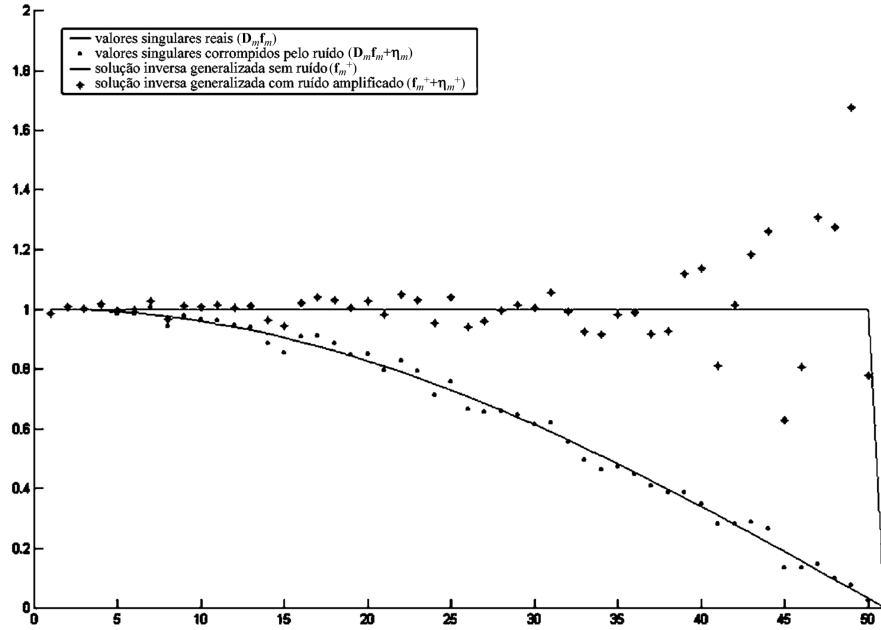


Figura 4.2: Valores singulares e perturbação devido ao ruído.

Segue-se, então, buscando a regularização do sistema para solucionar o problema do mal-condicionamento.

4.2.2. Solução regularizada

Conforme apresentado, a solução inversa generalizada, embora realizável para solucionar (4.1) ou (4.2), ainda é suscetível aos problemas do mal-condicionamento, levando à amplificação do ruído. Para resolver esse problema remanescente, é comum a utilização da *regularização*. Na regularização uma restrição é adicionada à otimização, com o que se busca uma resposta mais significativa para o sistema original [15,16]. Essa restrição é combinada de várias formas com a função objetivo, como:

$$\hat{\mathbf{f}}_m^{CLS} = \arg \min_{\mathbf{f}_m} \|\mathbf{C}\mathbf{f}_m\|_2, \text{ restrito à } \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2 = \varepsilon, \quad (4.13)$$

$$\hat{\mathbf{f}}_m^{CLS} = \arg \min_{\mathbf{f}_m} \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2, \text{ restrito à } \|\mathbf{C}\mathbf{f}_m\|_2 = E, \quad (4.14)$$

onde a sigla CLS significa *Constrained Least Squares*, ou Mínimos Quadrados com Restrição. O operador \mathbf{C} é uma função escolhida para compensar o efeito dos valores singulares pequenos,

impedindo que eles levem à uma desestabilização dos resultados, ou à ampliação de ruído. Em (4.13) e (4.14) são aplicadas restrições de igualdade com a finalidade de escolher soluções mais significativas para \mathbf{f}_m .

Podem-se, ainda, aplicar restrições de desigualdade, como:

$$\text{satisfazer } \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2 \leq \varepsilon, \text{ e também } \|\mathbf{C} \mathbf{f}_m\|_2 \leq E \quad (4.15)$$

Como qualquer das duas funções pode ser restrição ou função objetivo, é possível combinar quadraticamente as duas funções em uma única função sem restrição. Isso porque as funções utilizadas como restrição não são tão rígidas [16,17]. Dessa forma, consegue-se uma única função objetivo para o sistema, agora regularizada:

$$\mathbf{f}_m^{RLS} = \arg \min_{\mathbf{f}_m} \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2^2 + \lambda \|\mathbf{C} \mathbf{f}_m\|_2^2, \text{ onde } \lambda = (\varepsilon/E)^2, \quad (4.16)$$

onde a sigla RLS denota *Regularized Least Squares* ou Mínimos Quadrados Regularizados.

A Figura 4.3 representa, através de elipsóides, como seriam esses resultados para o caso bidimensional.

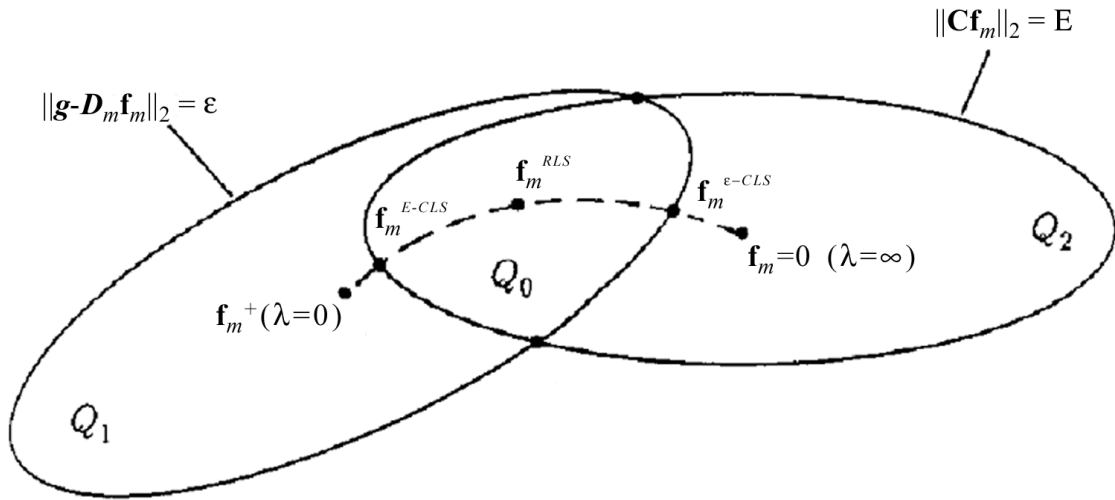


Figura 4.3: Representação das elipsóides para casos bidimensionais.

Geralmente, é utilizada como operador de regularização, \mathbf{C} , uma função passa-alta sem valores singulares nulos, o que levaria o mínimo de $\|\mathbf{C} \mathbf{f}_m\|_2$ ser igual a zero.

Na Figura 4.3, o conjunto de respostas Q_1 representa as respostas possíveis para $\|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2 \leq \varepsilon$ e o conjunto de respostas Q_2 representa as respostas possíveis para $\|\mathbf{C} \mathbf{f}_m\|_2 \leq E$. O conjunto de respostas Q_0 representa o conjunto que satisfaz o equacionamento em (4.15). O ponto $\mathbf{f}_m^{\varepsilon-CLS}$ é o mínimo que satisfaz a Equação (4.13) e o ponto \mathbf{f}_m^{E-CLS} é o mínimo que satisfaz a Equação (4.14). Sobre a linha pontilhada, encontram-se as respostas da minimização da Equação (4.16) para $0 \leq \lambda \leq \infty$. O ponto médio, \mathbf{f}_m^{RLS} , encontra-se em $\lambda = (\varepsilon/E)^2$.

É interessante notar na Figura 4.4 que tanto “ ε ” quanto “ E ” precisam ser conhecidos, estimados ou determinados de acordo com alguma informação *a priori* sobre o problema. Observando a Equação (4.3), pode-se concluir que escolher $\varepsilon \geq \|\eta\|_2$ é adequado, pois a elipsóide engloba a possível imagem original \mathbf{f}_m . Sendo o operador \mathbf{C} uma função que deve eliminar o ruído amplificado e preservar o sinal da imagem, deve-se escolher “ E ” de forma a incluir a imagem \mathbf{f}_m desejada dentro da elipsóide de regularização. Se o operador \mathbf{C} for, por exemplo, um passa-alta, é possível decidir que $E^2 \geq$ energia da versão da imagem \mathbf{f}_m filtrada por \mathbf{C} .

Se forem seguidos esses princípios, será obtida uma solução inversa regularizada com pouca ampliação de ruído e um resultado visual significativo. Todavia, a determinação de \mathbf{f}_m^{RLS} não é fácil, dado que é necessário solucionar:

$$(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}) \mathbf{f}_m = \mathbf{D}_m^T \mathbf{g}, \quad (4.17)$$

onde a solução é dada por

$$\hat{\mathbf{f}}_m^{RLS} = (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})^{-1} \mathbf{D}_m^T \mathbf{g} \quad (4.18)$$

A inversão da matriz em (4.18) é computacionalmente custosa [1,15-18,24] porque a matriz $(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})$ terá dimensão $M_1 M_2 \times M_1 M_2$ e, possivelmente, não é bloco-circulante. Em virtude do problema da inversão, opta-se pela busca iterativa do mínimo, que irá encontrar a cada passo um resultado mais próximo da Equação (4.18). A solução iterativa regularizada será apresentada no Capítulo 5.

Na próxima seção será vista mais detalhadamente a idéia de conjuntos teóricos, onde são considerados os problemas relacionados à concatenação da matriz \mathbf{D}_m e do vetor η quando as imagens estão sujeitas a ruídos com normas diferentes e erros de estimação.

4.3. Seqüência de Imagens e Conjuntos Teóricos

Nas seções anteriores, foi visto como o uso de seqüência de imagens pode melhorar o posto do sistema linear em (4.1). Com a concatenação das equações fornecidas pelas imagens da seqüência pode-se criar o sistema em (4.2), que possui um posto maior que o (4.1) e, conseqüentemente, um conjunto menor de soluções. Porém, a concatenação, na forma como foi apresentada, não é o método mais recomendado, visto que as características individuais de cada imagem são perdidas e ruídos com norma diferente em cada quadro e erros nas matrizes $\mathbf{D}_{k,m}$, devido a alguma estimação, são desconsiderados. Portanto, é necessário fazer uso de um método de combinação que respeite as contribuições individuais de cada imagem. A combinação por conjuntos teóricos [15-17] permite essa condição. Na seção seguinte o método será apresentado.

4.3.1. Combinando imagens diferentes e regularização

A solução apresentada pela Equação (4.16) é aplicável em situações em que todas as imagens da sequência têm a mesma norma de ruído e não contêm erros de estimação. Nas situações em que isso não acontece, é necessária a aplicação da devida ponderação em cada imagem. No método de conjuntos teóricos, os L sistemas lineares, fornecidos pelas L imagens da sequência, mais a restrição, são combinados quadraticamente, segundo (4.19).

$$\text{satisfazer:} \begin{cases} \|\mathbf{C}\mathbf{f}_m\|_2 \leq E \\ \|\mathbf{g}_1 - \mathbf{D}_{1,m}\mathbf{f}_m\|_2 \leq \varepsilon_1 \\ \vdots \\ \|\mathbf{g}_L - \mathbf{D}_{L,m}\mathbf{f}_m\|_2 \leq \varepsilon_L \end{cases} \quad (4.19)$$

Agora, cada imagem tem o próprio conjunto de soluções. Esses conjuntos teóricos formam elipsóides abertas, cujas bordas se estendem ao infinito nos eixos onde os valores singulares são nulos. Isso acontece porque os sistemas lineares fornecidos pelas imagens são subdeterminados, ou $p \leq N_1 N_2 \leq M_1 M_2$. Somente a função de regularização pode produzir uma elipsóide fechada, porque o operador \mathbf{C} pode ser escolhido para possuir todos os valores singulares reais e maiores que zero.

Para aproveitar a individualidade de cada imagem da sequência, ao invés da concatenação dos sistemas lineares como em (4.2), eles serão ponderados e combinados quadraticamente em uma nova função objetivo, conforme (4.20):

$$\hat{\mathbf{f}}_m^{BEM} = \arg \min_{\mathbf{f}_m} \frac{\kappa \|\mathbf{C}\mathbf{f}_m\|_2^2}{E^2} + \frac{\|\mathbf{g}_1 - \mathbf{D}_{1,m}\mathbf{f}_m\|_2^2}{\varepsilon_1^2} + \dots + \frac{\|\mathbf{g}_L - \mathbf{D}_{L,m}\mathbf{f}_m\|_2^2}{\varepsilon_L^2}, \quad (4.20)$$

onde a sigla BEM denota *Bounding Ellipsoidal Method*. Observando e comparando as Equações (4.16) e (4.20), vê-se que a função de otimização é diferente, mas mantém o equilíbrio de energias entre informação das imagens e restrição, se κ for escolhido para tal. Considera-se

$$\frac{\|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2^2}{\varepsilon^2} + \frac{\|\mathbf{C}\mathbf{f}_m\|_2^2}{E^2} \equiv \frac{\|\mathbf{g}_1 - \mathbf{D}_{1,m}\mathbf{f}_m\|_2^2}{\varepsilon_1^2} + \dots + \frac{\|\mathbf{g}_L - \mathbf{D}_{L,m}\mathbf{f}_m\|_2^2}{\varepsilon_L^2} + \frac{L \|\mathbf{C}\mathbf{f}_m\|_2^2}{E^2}, \quad (4.21)$$

onde $\varepsilon^2, \varepsilon_1^2, \dots, \varepsilon_L^2$ são os coeficientes de desigualdade em (4.15) e (4.19), geralmente escolhidos de acordo com $\varepsilon = \|\boldsymbol{\eta}\|_2$ e $\varepsilon_k = \|\boldsymbol{\eta}_k\|_2$, ou as normas dos vetores de ruído concatenado, $\boldsymbol{\eta}$, e de cada quadro, $\boldsymbol{\eta}_k$, respectivamente. Observa-se que, assim, κ tem de ser igual a L para que as equações tenham o mesmo balanço de influência entre imagens e restrição, visto que a elipsóide formada pela equação $\|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2^2 / \varepsilon^2 = 1$ terá de ter a mesma influência que a elipsóide formada pelas equações de cada imagem, $\|\mathbf{g}_k - \mathbf{D}_{k,m} \mathbf{f}_m\|_2^2 / \varepsilon_k^2 = 1$, somadas. Como são usadas L imagens, o fator de balanço deve ser $\kappa = L$. Pode-se modificar a Equação (4.20) de forma que se apresente como:

O exemplo também mostra a diferença das elipsóides formadas pelo método usando concatenação simples e pelo método ponderado. Nota-se que, com a concatenação simples, a matriz degradação \mathbf{D}_m passa a ter os dois valores singulares iguais, 0,5 e 0,5, para este exemplo, e a norma do vetor de ruído concatenado passa a ser $\|\boldsymbol{\eta}\|^2 = \|\boldsymbol{\eta}_1\|^2 + \|\boldsymbol{\eta}_2\|^2$. Assim, o sistema perde a contribuição individual de cada quadro da seqüência e a elipsóide passa a ser uma circunferência. Já, no método ponderado, a contribuição de cada imagem é preservada. Nessas condições, observa-se que a elipsóide formada (*Bounding Ellipsoid*) acompanha a forma da soma ponderada das elipsóides produzidas por cada sistema. Sem a regularização, as soluções (pontos centrais) são iguais para os dois métodos. Somente a forma das elipsóides, representada pelas curvas de nível das funções de otimização, é diferente.

Foi mencionado no começo da seção que é possível realizar o procedimento de concatenação e obter (4.20) se as equações forem ponderadas. Isso é feito segundo (4.24):

$$\begin{bmatrix} \frac{\mathbf{g}_1}{\varepsilon_1} \\ \vdots \\ \frac{\mathbf{g}_L}{\varepsilon_L} \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{D}_{1,m}}{\varepsilon_1} \\ \vdots \\ \frac{\mathbf{D}_{L,m}}{\varepsilon_L} \end{bmatrix} \mathbf{f}_m + \begin{bmatrix} \frac{\boldsymbol{\eta}_1}{\varepsilon_1} \\ \vdots \\ \frac{\boldsymbol{\eta}_L}{\varepsilon_L} \end{bmatrix}, \text{ ou } \mathbf{g} = \mathbf{D}_m \mathbf{f}_m + \boldsymbol{\eta}, \quad (4.24)$$

onde, agora, a equação concatenada está devidamente ponderada segundo o fator ε_k individual do sistema linear de cada imagem. As notações dos vetores \mathbf{g} e $\boldsymbol{\eta}$ normalizados e concatenados e da matriz \mathbf{D}_m normalizada e concatenada são as mesmas do sistema anterior, mas não deve haver preocupação quanto a isso, pois será utilizada esta nova matriz concatenada ponderada deste ponto em diante. Observa-se em (4.25) que a escolha do fator $\kappa=L$ se comprova, pois $\|\boldsymbol{\eta}\|^2 = \|\boldsymbol{\eta}_1\|^2/\varepsilon_1^2 + \dots + \|\boldsymbol{\eta}_L\|^2/\varepsilon_L^2 = L$, sendo escolhidos $\varepsilon_k^2 = \|\boldsymbol{\eta}_k\|^2$. Assim, a função de otimização pode ser escrita sem os somatórios, segundo (4.25):

$$\hat{\mathbf{f}}_m^{BEM} = \underset{\mathbf{f}_m}{\operatorname{argmin}} \lambda \|\mathbf{C}\mathbf{f}_m\|_2^2 + \sum_{k=1}^L \varepsilon_k^{-2} \|\mathbf{g}_k - \mathbf{D}_{k,m} \mathbf{f}_m\|_2^2 = \underset{\mathbf{f}_m}{\operatorname{argmin}} \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|_2^2 + \lambda \|\mathbf{C}\mathbf{f}_m\|_2^2, \text{ onde } \lambda = \frac{\kappa}{E^2} \quad (4.25)$$

A solução é apresentada em (4.26). Esta seria igual a (4.18) se as ponderações ε_k fossem iguais.

$$\hat{\mathbf{f}}_m^{BEM} = (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})^{-1} \mathbf{D}_m^T \mathbf{g} \quad (4.26)$$

Agora o sistema representa melhor a contribuição de cada imagem, seja por ruído com diferentes normas, seja por erros na estimação das matrizes $\mathbf{D}_{k,m}$. A igualdade apresentada na Equação (4.25) é facilmente comprovada através de (4.27a-b).

$$\mathbf{D}_m^T \mathbf{D}_m = \begin{bmatrix} \mathbf{D}_{1,m} \\ \varepsilon_1 \\ \vdots \\ \mathbf{D}_{L,m} \\ \varepsilon_L \end{bmatrix}^T \begin{bmatrix} \mathbf{D}_{1,m} \\ \varepsilon_1 \\ \vdots \\ \mathbf{D}_{L,m} \\ \varepsilon_L \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{1,m}^T & \dots & \mathbf{D}_{L,m}^T \\ \varepsilon_1 & & \varepsilon_L \end{bmatrix} = \sum_{k=1}^L \varepsilon_k^{-2} \mathbf{D}_{k,m}^T \mathbf{D}_{k,m} \quad (4.27a)$$

$$\mathbf{D}_m^T \mathbf{g} = \begin{bmatrix} \mathbf{D}_{1,m} \\ \varepsilon_1 \\ \vdots \\ \mathbf{D}_{L,m} \\ \varepsilon_L \end{bmatrix}^T \begin{bmatrix} \mathbf{g}_1 \\ \varepsilon_1 \\ \vdots \\ \mathbf{g}_L \\ \varepsilon_L \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{1,m}^T & \dots & \mathbf{D}_{L,m}^T \\ \varepsilon_1 & & \varepsilon_L \end{bmatrix} \begin{bmatrix} \mathbf{g}_1 \\ \varepsilon_1 \\ \vdots \\ \mathbf{g}_L \\ \varepsilon_L \end{bmatrix} = \sum_{k=1}^L \varepsilon_k^{-2} \mathbf{D}_{k,m}^T \mathbf{g}_k \quad (4.27b)$$

O motivo pelo qual é interessante ter-se a devida ponderação revela-se na Figura 4.6, onde é ilustrado o efeito da adição da restrição.

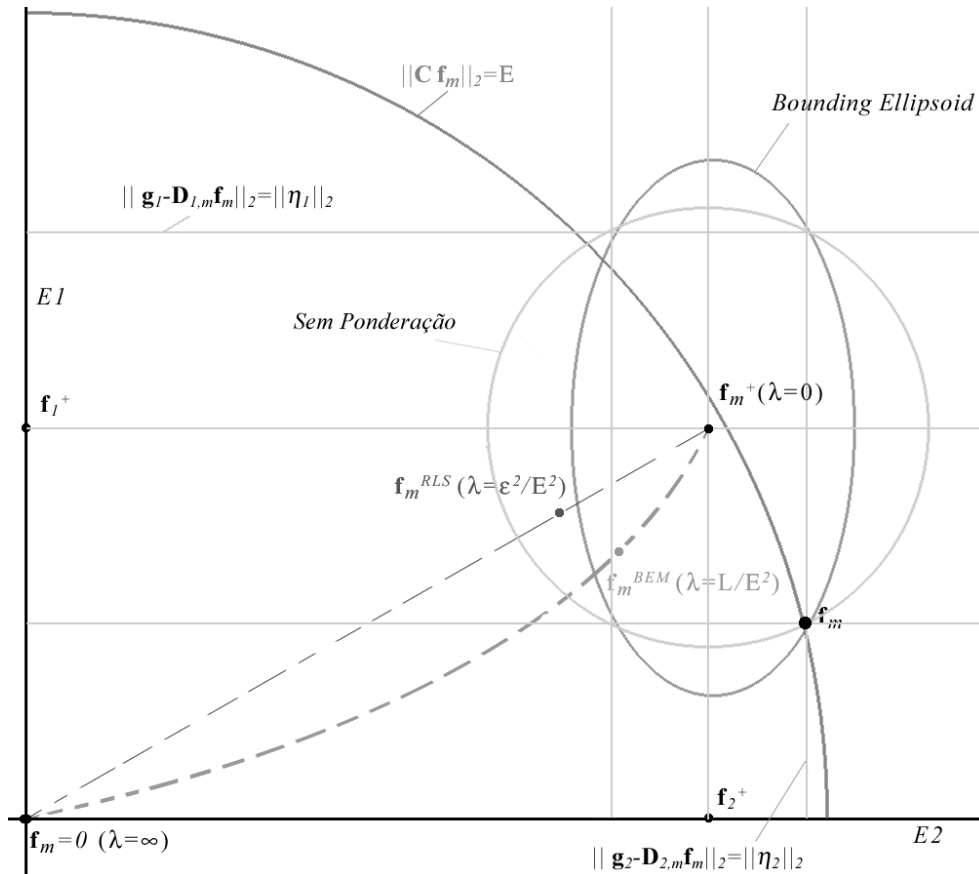


Figura 4.6: Elipsóides geradas pelas imagens em combinação com a restrição.

Através da Figura 4.6 pode-se observar que o comportamento das soluções quando a restrição é adicionada nas funções objetivo é diferente. As linhas pontilhadas sobre as quais estão \mathbf{f}_m^{RLS} e \mathbf{f}_m^{BEM} são as respostas para as respectivas soluções das Equações (4.18) e (4.26), variando-se o λ de 0 a ∞ . Nota-se que, como a influência da equação $\|\mathbf{g}_1 - \mathbf{D}_{1,m} \mathbf{f}_m\|_2 = \|\boldsymbol{\eta}_1\|_2$ é menor, por ter norma do ruído maior, a linha de solução que contém \mathbf{f}_m^{BEM} desloca-se mais rapidamente pelo eixo da equação $\|\mathbf{g}_1 - \mathbf{D}_{1,m} \mathbf{f}_m\|_2 = \|\boldsymbol{\eta}_1\|_2$.

Dessa forma, o sistema em (4.24) expressa melhor a contribuição dos quadros da sequência que o sistema em (4.2), portanto, é mais adequado para ser utilizado no método de Mínimos Quadrados Regularizados.

Como foi visto, o uso de conjuntos teóricos não é um método de otimização, mas uma forma de combinar as imagens através de várias funções restrições. Como as funções ainda seguem o método de Mínimos Quadrados e ainda utilizam regularização, as equações que contêm a sigla BEM (*Bounding ellipsoidal Method*) serão chamadas de equações de Mínimos Quadrados Regularizados, usando novamente a sigla RLS (*Regularized Least Squares*). Os trabalhos [10,15-17] utilizam o método de conjuntos teóricos e o mencionam como método de Mínimos Quadrados Regularizados.

Até agora foi levado em conta nos exemplos apenas o ruído para a ponderação de cada imagem. No Capítulo 6, onde a estimação de movimento será apresentada, o ponderador ϵ_k recomendado englobará, além do ruído, o erro de estimação, como em [10]. No Capítulo 7, apresentam-se as escolhas práticas do fator de regularização, λ , e das ponderações, ϵ_k , na presença de *outliers*.

Como em (4.18), a matriz $(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})$ da Equação (4.26) é de difícil inversão por causa de sua elevada dimensão ($M_1 M_2 \times M_1 M_2$). Portanto, a busca do ponto ótimo em (4.26) deve ser realizada na forma iterativa. O processo iterativo será visto no Capítulo 5.

Conclusões do Capítulo 4

Neste capítulo foi vista a trajetória de desenvolvimento das soluções determinísticas para os sistemas lineares, mencionando-se os problemas da solução de sistemas e direcionando para as soluções de Mínimos Quadrados.

A necessidade dos processos de otimização e regularização para encontrar a solução ótima para sistemas com sequência de imagens foi explicada. Explicou-se e ilustrou-se a demonstração da vantagem da concatenação ponderada através de conjuntos teóricos, que foi adotada neste trabalho. No Capítulo 7 serão incorporadas modificações a essas equações para uso em sequência de imagens com *outliers*.

Capítulo 5: Solução Iterativa Regularizada

Neste capítulo, apresenta-se a busca iterativa da solução inversa da degradação. Conforme a teoria do Capítulo 4, o método de Mínimos Quadrados Regularizados é utilizado para solucionar a degradação através da otimização da função objetivo apresentada na Equação (5.1).

$$\hat{\mathbf{f}}_m^{RLS} = \arg \min_{\mathbf{f}_m} \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m\|^2 + \lambda \|\mathbf{C} \mathbf{f}_m\|^2, \text{ onde } \lambda = \kappa/E^2 \quad (5.1)$$

A solução é dada por:

$$\hat{\mathbf{f}}_m^{RLS} = (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})^{-1} \mathbf{D}_m^T \mathbf{g} \quad (5.2)$$

A função em (5.1) deve ser *estritamente convexa*, o que significa que a matriz $(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})$ será sempre *definida positiva* se o operador \mathbf{C} realizar a regularização adequada. A matriz definida positiva tem todos autovalores maiores que zero [4-5]. Dessa forma, a solução ótima será global e única; as linhas de contorno ou de nível da função objetivo formarão elipsóides fechadas. Assim, o problema de otimização poderá fazer uso dos métodos de *Projeção de Gradiente* para encontrar a solução da equação. Em seguida, mostram-se os métodos de projeção de gradiente comumente encontrados na literatura para a solução iterativa regularizada com seqüência de imagens.

5.1. Soluções Iterativas de Projeção de Gradiente

Em aplicações nas quais as matrizes têm um tamanho significativo, a inversão dessas pode ser computacionalmente custosa. Uma boa forma de solucionar o problema é usar os métodos iterativos, dentre os quais o mais aplicado é o de *Aproximações Sucessivas* [1]. A Equação (5.3) apresenta um exemplo onde n é a iteração corrente.

$$\mathbf{f}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta \Phi(\mathbf{f}_m^{(n)}) \quad (5.3)$$

Em virtude de o problema de otimização em (5.1) formar um conjunto convexo, pode-se fazer uso dos métodos de Projeção de Gradiente para encontrar a solução. Nesses métodos, o passo de aproximação, $\Phi(\mathbf{f}_m^{(n)})$, é dado por:

$$\Phi(\mathbf{f}_m^{(n)}) = -\nabla \mathbf{F}(\mathbf{f}_m^{(n)}), \quad (5.4)$$

onde $\mathbf{F}(\mathbf{f}_m^{(n)})$ é a função objetivo, ou de minimização, apresentada na Equação (5.5).

$$\mathbf{F}(\mathbf{f}_m^{(n)}) = \|\mathbf{g} - \mathbf{D}_m \mathbf{f}_m^{(n)}\|^2 + \lambda \|\mathbf{C} \mathbf{f}_m^{(n)}\|^2 \quad (5.5)$$

O negativo do gradiente da função de minimização é mostrado em (5.6).

$$-\nabla \mathbf{F}(\mathbf{f}_m^{(n)}) = \mathbf{D}_m^T \mathbf{g} - (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}) \mathbf{f}_m^{(n)} \quad (5.6)$$

Normalmente, as iterações iniciam com $\mathbf{f}_m^{(0)}=0$ e, se β garantir a convergência, esta será atingida quando $\Phi(\mathbf{f}_m^{(n)})=0$. Então, $\nabla\Phi(\mathbf{f}_m^{(n)})=0$ é o ponto fixo da iteração ou solução do sistema.

Embora a solução iterativa seja, aparentemente, mais complexa que uma simples inversão de matrizes, ela traz vantagens significativas, como:

1. obter a solução inversa $(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})^{-1} \mathbf{D}_m^T \mathbf{g}$ sem computar a matriz inversa;
2. o resultado poder ser monitorado ao longo das iterações;
3. as operações não precisam, necessariamente, ser executadas na forma matricial (não tornando necessário a criação das matrizes, apenas a aplicação das operações referentes em outros domínios onde pode ser computacionalmente menos custoso);
4. em alguns casos, a convergência é rápida, tornando possíveis a utilização em aplicações que exijam rapidez na resposta (através do uso de métodos com alta taxa de convergência);
5. providenciar naturalmente uma transição de uma imagem com pouco detalhe para uma imagem com máximo nível de detalhe (solução ótima) ao longo das iterações.

A seguir, são apresentadas duas formas bem conhecidas de iteração de aproximação sucessivas que fazem uso da Projeção do Gradiente. Serão vistos o método de Distanciamento Simultâneo (*Simultaneous Displacement*), onde a análise da convergência é demonstrada e o método de Descida mais Íngreme (*Steepest Descent*), onde o fator de relaxamento β pode garantir uma taxa de convergência maior, demonstrando a possibilidade do uso de iterações com alta taxa de convergência.

5.1.1. Iteração de Distanciamento Simultâneo

A iteração de Distanciamento Simultâneo é uma das formas mais simples e menos exigentes, no ponto de vista computacional, de busca da solução. Nesta técnica, β assume um valor fixo que leva a imagem \mathbf{f}_m a convergir para $\mathbf{f}_m^{\text{RLS}}$. O β é fixo, ou seja, terá sempre o mesmo valor em todas as iterações.

O método será analisado segundo sua convergência. Substituindo a Equação (5.6) em (5.4) e, novamente, substituindo em (5.3), obtém-se:

$$\begin{aligned} \mathbf{f}_m^{(0)} &= 0, & \mathbf{f}_m^{(1)} &= \beta \mathbf{D}_m^T \mathbf{g} \\ \mathbf{f}_m^{(n+1)} &= \mathbf{f}_m^{(n)} + \beta (\mathbf{D}_m^T \mathbf{g} - (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}) \mathbf{f}_m^{(n)}) \end{aligned} \quad (5.7)$$

Para continuar a análise, a Equação (5.7) é reduzida conforme:

$$\begin{aligned} \mathbf{f}_m^{(n+1)} &= \beta \mathbf{D}_m^T \mathbf{g} + (\mathbf{I} - \beta (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})) \mathbf{f}_m^{(n)} \\ \mathbf{f}_m^{(n+1)} &= \mathbf{f}_m^{(1)} + (\mathbf{I} - \beta \mathbf{B}) \mathbf{f}_m^{(n)} \\ \mathbf{f}_m^{(n+1)} &= \mathbf{f}_m^{(1)} + (\mathbf{A}) \mathbf{f}_m^{(n)}, \end{aligned} \quad (5.8)$$

onde \mathbf{I} é a matriz identidade, a matriz $\mathbf{B}=(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})$ e a matriz $\mathbf{A}=(\mathbf{I}-\beta \mathbf{B})$. A Equação (5.8) demonstra que o processo iterativo age como um sistema recursivo de múltiplas entradas e múltiplas saídas. Portanto, sendo a entrada $\mathbf{f}_m^{(1)}$, a saída na iteração n pode ser calculada como:

$$\begin{aligned}\mathbf{f}_m^{(n)} &= \mathbf{A} \mathbf{f}_m^{(n-1)} + \mathbf{f}_m^{(1)} \\ \mathbf{f}_m^{(n)} &= \sum_{i=1}^n \mathbf{A}^i \mathbf{f}_m^{(1)} = \frac{\mathbf{I} - \mathbf{A}^{n+1}}{\mathbf{I} - \mathbf{A}} \mathbf{f}_m^{(1)},\end{aligned}\quad (5.9)$$

onde a divisão por matriz significa multiplicação pela matriz inversa e é adotada para simplificar as equações seguintes. Maiores informações sobre essas deduções são encontradas em [1,6]. Retornando a redução da Equação (5.8) na expressão em (5.9), tem-se:

$$\begin{aligned}\mathbf{f}_m^{(n)} &= \mathbf{A} \mathbf{f}_m^{(n-1)} + \mathbf{f}_m^{(1)} = (\mathbf{I} - \beta(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})) \mathbf{f}_m^{(n-1)} + \beta \mathbf{D}_m^T \mathbf{g} \\ \mathbf{f}_m^{(n)} &= \frac{\mathbf{I} - \mathbf{A}^{n+1}}{\mathbf{I} - \mathbf{A}} \mathbf{f}_m^{(1)} = \frac{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}))^{n+1}}{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}))} \beta \mathbf{D}_m^T \mathbf{g}\end{aligned}\quad (5.10)$$

Para continuar a análise da convergência, é necessária a aplicação da decomposição em valores singulares, determinada por (5.11a-b).

$$\mathbf{D}_m = \mathbf{U}_D \mathbf{S}_D \mathbf{V}_D^T = \sum_{i=1}^p \sigma_{Di} \mathbf{u}_{Di} \mathbf{v}_{Di}^T \quad (5.11a)$$

$$\mathbf{C} = \mathbf{U}_C \mathbf{S}_C \mathbf{V}_C^T = \sum_{i=1}^q \sigma_{Ci} \mathbf{u}_{Ci} \mathbf{v}_{Ci}^T$$

$$\mathbf{D}_m^T \mathbf{D}_m = \mathbf{V}_D \mathbf{S}_D^T \mathbf{S}_D \mathbf{V}_D^T = \sum_{i=1}^p \mathbf{v}_{Di} \sigma_{Di}^2 \mathbf{v}_{Di}^T \quad (5.11b)$$

$$\mathbf{C}^T \mathbf{C} = \mathbf{V}_C \mathbf{S}_C^T \mathbf{S}_C \mathbf{V}_C^T = \sum_{i=1}^q \mathbf{v}_{Ci} \sigma_{Ci}^2 \mathbf{v}_{Ci}^T$$

Aplicando a decomposição (5.11a) e (5.11b) em (5.10), obtém-se:

$$\mathbf{f}_m^{(n)} = \frac{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{V}_D \mathbf{S}_D^T \mathbf{S}_D \mathbf{V}_D^T + \lambda \mathbf{V}_C \mathbf{S}_C^T \mathbf{S}_C \mathbf{V}_C^T))^{n+1}}{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{V}_D \mathbf{S}_D^T \mathbf{S}_D \mathbf{V}_D^T + \lambda \mathbf{V}_C \mathbf{S}_C^T \mathbf{S}_C \mathbf{V}_C^T))} \beta \mathbf{V}_D \mathbf{S}_D \mathbf{U}_D^T \mathbf{g} \quad (5.12)$$

Considerando que os vetores singulares de \mathbf{D}_m (autovetores de $\mathbf{D}_m^T \mathbf{D}_m$) e de \mathbf{C} são iguais, ou seja, $\mathbf{V}_D = \mathbf{V}_C = \mathbf{V}$ e $\mathbf{U}_D = \mathbf{U}_C = \mathbf{U}$, a Expressão de (5.12) pode ser reduzir ainda mais, chegando à expressão em (5.13). Informações sobre as propriedades da SVD podem ser encontradas em [4].

$$\begin{aligned}\mathbf{f}_m^{(n)} &= \mathbf{V} \frac{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C))^{n+1}}{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C))} \beta \mathbf{S}_D \mathbf{U}^T \mathbf{g} \\ \mathbf{f}_m^{(n)} &= \sum_{i=1}^q \mathbf{v}_i \frac{1 - (1 - \beta(\sigma_{Di}^2 + \lambda \sigma_{Ci}^2))^{n+1}}{1 - (1 - \beta(\sigma_{Di}^2 + \lambda \sigma_{Ci}^2))} \beta \sigma_{Di} \mathbf{u}_i^T \mathbf{g}\end{aligned}\quad (5.13)$$

Agora, pode-se explicar, de forma mais clara, o conceito de convergência. Observe-se em (5.13) que o termo que é elevado à potência de $n+1$ deve convergir ou ser estável. Para convergência é necessário que a matriz \mathbf{A} seja contrativa, o que significa que a norma da matriz, ou seu maior valor singular, é menor que um [15-17], conforme (5.14).

$$\|\mathbf{A}\|_2 < 1, \text{ ou } \|\mathbf{I} - \beta\mathbf{B}\|_2 < 1, \text{ ou ainda } \|1 - \beta(\sigma_{D1}^2 + \lambda\sigma_{C1}^2)\|_2 < 1, \quad (5.14)$$

onde σ_{D1}^2 é o primeiro e maior autovalor da matriz $\mathbf{D}_m^T \mathbf{D}_m$ e σ_{C1}^2 é o autovalor da matriz regularizadora $\mathbf{C}^T \mathbf{C}$ no mesmo autovetor de σ_{D1}^2 . Pela definição em [4], a norma, $\|\cdot\|_2$, de uma matriz é igual ao seu maior valor singular.

Se essa condição ocorrer, no limite em que as iterações $n \rightarrow \infty$, a resposta será:

$$\begin{aligned} \mathbf{f}_m^{(\infty)} &= \mathbf{V} \frac{\mathbf{S}_D \mathbf{U}^T \mathbf{g}}{(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C)} \\ \mathbf{f}_m^{(\infty)} &= \sum_{i=1}^q \mathbf{v}_i \frac{\sigma_{Di} \mathbf{u}_i^T \mathbf{g}}{(\sigma_{Di}^2 + \lambda \sigma_{Ci}^2)} \end{aligned} \quad (5.15)$$

Observa-se que é a mesma resposta dada pela Equação (5.2), porém expressa em função de seus valores singulares e vetores singulares. Mais detalhes sobre isso também serão vistos na Seção 5.2.1. Vale lembrar que, se a matriz \mathbf{D}_m fosse bloco circulante, os valores singulares seriam os coeficientes da Transformada 2D de Fourier.

Por último, o valor de β necessário para convergência, extraído de (5.14), fica entre:

$$0 < \beta < \frac{2}{\sigma_{D1}^2 + \lambda \sigma_{C1}^2} \quad (5.16)$$

Pode-se utilizar também a norma de matrizes, como em (5.17).

$$0 < \beta < \frac{2}{\|\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}\|_2} \quad (5.17)$$

Utilizando o β correto, o método iterativo de Distanciamento Simultâneo pode ser aplicado com garantias de convergência. O valor de β também controla a taxa de convergência, conseguindo que se chegue mais rapidamente no resultado desejado. A seguir, apresenta-se um método onde o β é escolhido de forma a acelerar a convergência.

5.1.2. Iteração de Descida mais Íngreme

O método de Descida mais Íngreme (*Steepest Descent*) é caracterizado por utilizar um β diferente a cada iteração. Utiliza-se o valor de β que traz a melhor convergência na iteração n em questão. Observe-se na Figura 5.1, que demonstra o caso bidimensional. Sendo $\Phi(\mathbf{f}_m^{(n)})$ a direção de busca atual, direção esta que é dada pelo gradiente da função objetivo no ponto $\mathbf{f}_m^{(n)}$, β deve garantir que o próximo ponto, $\mathbf{f}_m^{(n+1)}$, seja o mínimo na direção $\Phi(\mathbf{f}_m^{(n)})$. Como não se conhece o ponto mínimo nesta direção, assume-se que o mínimo ocorre quando a próxima direção de busca, $\Phi(\mathbf{f}_m^{(n+1)})$, é ortogonal à atual [7].

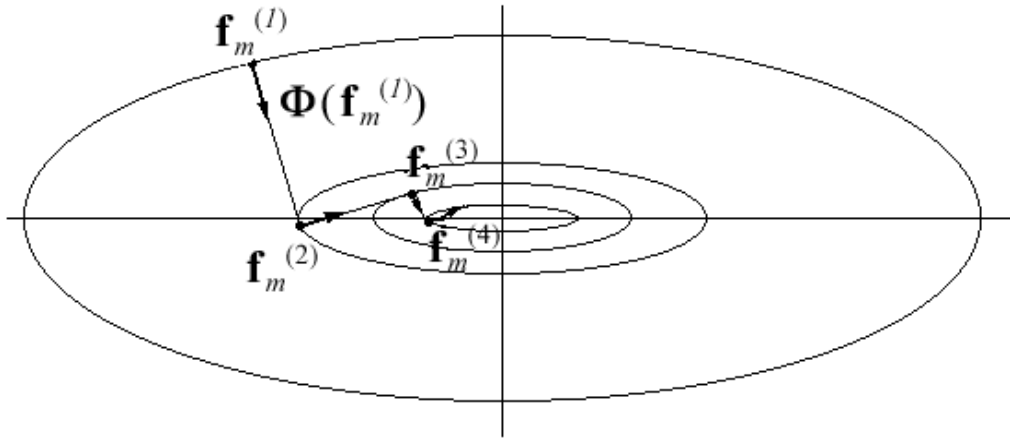


Figura 5.1: Busca pelo mínimo no caso Descida mais Íngreme.

A Figura 5.1 demonstra que a direção atual de busca é o negativo do gradiente da função de objetivo, ou $-\nabla F(\mathbf{f}_m^{(n)})$. O ponto seguinte é o que tem direção de busca ortogonal a direção atual, embora isso não signifique que qualquer uma das direções de busca seguintes, $\Phi(\mathbf{f}_m^{(p)})$ onde $p > n+1$, seja ortogonal à direção atual. Dessa forma, deve-se determinar o β para que $\langle \Phi(\mathbf{f}_m^{(n)}), \Phi(\mathbf{f}_m^{(n+1)}) \rangle = 0$, onde $\langle \mathbf{x}, \mathbf{y} \rangle$ é produto interno dos vetores \mathbf{x} e \mathbf{y} .

Antes da análise da determinação de β , será observado como as iterações na Expressão (5.18) procedem, de forma que as projeções possam ser relacionadas.

$$\begin{aligned} \mathbf{f}_m^{(1)} &= \mathbf{f}_m^{(0)} + \beta^{(0)} \Phi(\mathbf{f}_m^{(0)}) \\ &\vdots \\ \mathbf{f}_m^{(n+1)} &= \mathbf{f}_m^{(n)} + \beta^{(n)} \Phi(\mathbf{f}_m^{(n)}) \\ \mathbf{f}_m^{(n+2)} &= \mathbf{f}_m^{(n+1)} + \beta^{(n+1)} \Phi(\mathbf{f}_m^{(n+1)}) \end{aligned} \quad (5.18)$$

Pode-se observar em (5.18) que as projeções são relacionadas, como:

$$\Phi(\mathbf{f}_m^{(n+1)}) = \Phi(\mathbf{f}_m^{(n)} + \beta^{(n)} \Phi(\mathbf{f}_m^{(n)})) \quad (5.19)$$

Substituindo a direção de busca pelo negativo do gradiente da função objetivo, obtém-se:

$$\begin{aligned} \Phi(\mathbf{f}_m^{(n)}) &= -\nabla F(\mathbf{f}_m^{(n)}) = \mathbf{D}_m^T \mathbf{g} - (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}) \mathbf{f}_m^{(n)} \\ \Phi(\mathbf{f}_m^{(n)}) &= \mathbf{b} - \mathbf{B} \mathbf{f}_m^{(n)}, \quad \text{onde } \mathbf{b} = \mathbf{D}_m^T \mathbf{g} \quad \text{e} \quad \mathbf{B} = (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}) \end{aligned} \quad (5.20)$$

Então, as projeções ou direções de busca podem ser relacionadas segundo (5.21), através das substituições de (5.20) e (5.19) em (5.18).

$$\begin{aligned} \Phi(\mathbf{f}_m^{(n+1)}) &= \mathbf{b} - \mathbf{B} \mathbf{f}_m^{(n+1)} \\ \Phi(\mathbf{f}_m^{(n+1)}) &= \mathbf{b} - \mathbf{B}(\mathbf{f}_m^{(n)} + \beta^{(n)} \Phi(\mathbf{f}_m^{(n)})) \\ \Phi(\mathbf{f}_m^{(n+1)}) &= \mathbf{b} - \mathbf{B} \mathbf{f}_m^{(n)} - \beta^{(n)} \mathbf{B} \Phi(\mathbf{f}_m^{(n)}) \\ \Phi(\mathbf{f}_m^{(n+1)}) &= \Phi(\mathbf{f}_m^{(n)}) - \beta^{(n)} \mathbf{B} \Phi(\mathbf{f}_m^{(n)}) \end{aligned} \quad (5.21)$$

Observa-se que a projeção seguinte pode ser produzida com a projeção atual, desde que se conheça o β correto. Assim, se projeções ortogonais são desejadas, β deve ser escolhido de acordo com (5.22).

$$\begin{aligned}
 \langle \Phi(\mathbf{f}_m^{(n+1)}), \Phi(\mathbf{f}_m^{(n)}) \rangle &= 0 \\
 \langle (\Phi(\mathbf{f}_m^{(n)}) - \beta^{(n)} \mathbf{B} \Phi(\mathbf{f}_m^{(n)})), \Phi(\mathbf{f}_m^{(n)}) \rangle &= 0 \\
 \langle \Phi(\mathbf{f}_m^{(n)}), \Phi(\mathbf{f}_m^{(n)}) \rangle &= \langle \beta^{(n)} \mathbf{B} \Phi(\mathbf{f}_m^{(n)}), \Phi(\mathbf{f}_m^{(n)}) \rangle \\
 \beta^{(n)} &= \frac{\langle \Phi(\mathbf{f}_m^{(n)}), \Phi(\mathbf{f}_m^{(n)}) \rangle}{\langle \mathbf{B} \Phi(\mathbf{f}_m^{(n)}), \Phi(\mathbf{f}_m^{(n)}) \rangle}
 \end{aligned} \tag{5.22}$$

Dessa forma, o método de busca de Descida mais Íngreme torna-se mais eficiente que o método de Distanciamento Simultâneo, encontrando o \mathbf{f}_m mínimo mais rapidamente. O conjunto de equações utilizadas na técnica de Descida mais Íngreme está resumido em (5.23a-d):

$$\Phi(\mathbf{f}_m^{(0)}) = -\nabla \mathbf{F}(\mathbf{f}_m^{(0)}) = \mathbf{D}_m^T \mathbf{g} - \mathbf{B} \mathbf{f}_m^{(0)} = \mathbf{D}_m^T \mathbf{g}, \quad \text{onde } \mathbf{f}_m^{(0)} = 0 \quad \text{e} \quad \mathbf{B} = (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}) \tag{5.23a}$$

$$\beta^{(n)} = \frac{\langle \Phi(\mathbf{f}_m^{(n)}), \Phi(\mathbf{f}_m^{(n)}) \rangle}{\langle \mathbf{B} \Phi(\mathbf{f}_m^{(n)}), \Phi(\mathbf{f}_m^{(n)}) \rangle} \tag{5.23b}$$

$$\mathbf{f}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta^{(n)} \Phi(\mathbf{f}_m^{(n)}) \tag{5.23c}$$

$$\Phi(\mathbf{f}_m^{(n+1)}) = \Phi(\mathbf{f}_m^{(n)}) - \beta^{(n)} \mathbf{B} \Phi(\mathbf{f}_m^{(n)}) \tag{5.23d}$$

A convergência não pode ser demonstrada como no método de iteração anterior. Segundo [7], o método de Descida mais Íngreme pode falhar quanto à convergência, principalmente quando muito próximo do mínimo, caso haja uma grande diferença de autovalores na matriz $\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}$.

5.2. Propriedades Computacionais do Método Iterativo

Nesta seção, são demonstradas algumas propriedades do cálculo da imagem \mathbf{f}_m^{RLS} , na Equação (5.2), através da forma iterativa. Estas propriedades são:

- a não-necessidade de inversão da matriz $(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})$;
- a possibilidade de executar as operações em outro domínio, que não o matricial.

Essas propriedades levam o método iterativo a ter vantagens computacionais sobre a inversão da matriz. Essas propriedades permitem que a restauração da resolução possa ser usada em aplicações nas quais as imagens têm um tamanho significativo e o cálculo das matrizes requer alto custo computacional. Comentam-se essas propriedades nas duas seções seguintes.

5.2.1. Inversão da degradação

De acordo com a Equação (5.1), a busca da solução leva ao resultado na Equação (5.2), significando que, se o método de busca convergir para a solução estará invertendo a matriz $(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})$. Sabe-se que essa matriz tem dimensões $M_1 M_2 \times M_1 M_2$, o que em uma imagem modesta, com 320×320 pixels, por exemplo, significa inverter uma matriz de tamanho 102400×102400 . Dessa forma, o procedimento iterativo realizará a inversão naturalmente e com menor custo computacional [10,15-17].

A análise de convergência do método de busca de Distanciamento Simultâneo mostrou, através da Equação (5.15), que, se β garantir a convergência, \mathbf{f}_m convergirá para a imagem \mathbf{f}_m^{RLS} . Pela observação dos resultados da Equação (5.15) e da Equação (5.2), pode-se notar que, se a Equação (5.15) for retornada ao domínio dos valores singulares, o resultado será igual ao da Equação (5.2), o que é demonstrado de (5.24) a (5.25).

$$\mathbf{f}_m^{(n)} = \mathbf{V} \frac{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C))^{n+1}}{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C))} \beta \mathbf{S}_D \mathbf{U}^T \mathbf{g}, \quad (5.24)$$

onde $\mathbf{S}_A = \mathbf{I} - \beta(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C)$ é uma matriz diagonal com os valores singulares da matriz \mathbf{A} , definida em (5.8); \mathbf{I} é a matriz identidade; β , um escalar; \mathbf{S}_D , uma matriz diagonal composta pelos valores singulares de \mathbf{D}_m , e \mathbf{S}_C , outra matriz diagonal composta pelos valores singulares de \mathbf{C} .

Se β garantir convergência, tem-se que $\|\mathbf{A}\|_2 = \|\mathbf{S}_A\|_2 < 1$. Então, as matrizes \mathbf{A} e \mathbf{S}_A são contrativas, ou seja, $\|\mathbf{S}_A\|_2 > \|(\mathbf{S}_A)^2\|_2 > \dots > \|(\mathbf{S}_A)^\infty\|_2 = 0$. Dessa forma, no limite, quando $n \rightarrow \infty$, a matriz $(\mathbf{S}_A)^{n+1}$ tenderá a uma matriz de zeros. O resultado é observado em (5.25a-b).

$$\begin{aligned} \mathbf{f}_m^{(\infty)} &= \mathbf{V} \frac{\mathbf{I}}{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C))} \beta \mathbf{S}_D \mathbf{U}^T \mathbf{g} \\ \mathbf{f}_m^{(\infty)} &= \mathbf{V} \frac{\beta \mathbf{S}_D}{\mathbf{I} - (\mathbf{I} - \beta(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C))} \mathbf{U}^T \mathbf{g} \end{aligned} \quad (5.25a)$$

$$\begin{aligned} \mathbf{f}_m^{(\infty)} &= \mathbf{V} \frac{\beta \mathbf{S}_D}{\beta(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C)} \mathbf{U}^T \mathbf{g} \\ \mathbf{f}_m^{(\infty)} &= \mathbf{V} \frac{\mathbf{S}_D}{(\mathbf{S}_D^T \mathbf{S}_D + \lambda \mathbf{S}_C^T \mathbf{S}_C)} \mathbf{U}^T \mathbf{g} = \hat{\mathbf{f}}_m^{RLS} = \frac{\mathbf{D}_m^T}{(\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})} \end{aligned} \quad (5.25b)$$

Isso comprova a inversão. Embora, teoricamente, sejam necessárias infinitas interações para conseguir o resultado, na prática, com um número suficiente de iterações com boa convergência já é observado um resultado significativo. Por essa razão, os métodos com maior convergência conseguem melhores resultados com menos iterações, mesmo que não se confirme que eles podem convergir para \mathbf{f}_m^{RLS} , como é o caso do método de Descida mais Íngreme.

5.2.2. Implementação com menor custo computacional

A segunda propriedade do método iterativo, que se traduz em vantagem computacional, é a não-necessidade de realizar os cálculos no domínio matricial, significando que não é necessário “criar” as matrizes, da forma vista no Capítulo 2. De fato, precisa-se apenas aplicar o cálculo referente à transformação linear representada pela matriz. Isso significa que as matrizes $\mathbf{D}_{m,k}$, de tamanho $N_1N_2 \times M_1M_2$, e a matriz concatenada \mathbf{D}_m , de tamanho $LN_1N_2 \times M_1M_2$, não são usadas, ao invés, as operações que as matrizes representam podem ser realizadas. Por exemplo, na operação $\mathbf{A}_{k,m}\mathbf{f}_m$ é possível a aplicação de compensação de movimento, no lugar da multiplicação por \mathbf{H}_k , a convolução é aplicável e na operação \mathbf{S}_k , a extração dos *pixels* pode ser usada.

Antes de entrar nos detalhes de cada operação, a função de objetivo da otimização terá sua forma modificada para que fique mais fácil observar os pontos comentados. A iteração base dos métodos de projeção de gradiente é:

$$\mathbf{f}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta(\mathbf{D}_m^T \mathbf{g} - (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C}) \mathbf{f}_m^{(n)}) \quad (5.26)$$

Observa-se que a matriz $\mathbf{D}_m^T \mathbf{D}_m$ e o vetor $\mathbf{D}_m^T \mathbf{g}$ são obtidos pela multiplicação das matrizes e vetores concatenados, apresentados no Capítulo 4. Agora, observe-se a equação:

$$\mathbf{D}_m^T \mathbf{D}_m = \begin{bmatrix} \frac{\mathbf{D}_{1,m}}{\varepsilon_1} \\ \vdots \\ \frac{\mathbf{D}_{L,m}}{\varepsilon_L} \end{bmatrix}^T \begin{bmatrix} \frac{\mathbf{D}_{1,m}}{\varepsilon_1} \\ \vdots \\ \frac{\mathbf{D}_{L,m}}{\varepsilon_L} \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{1,m}^T & \dots & \mathbf{D}_{L,m}^T \\ \varepsilon_1 & & \varepsilon_L \end{bmatrix} \begin{bmatrix} \frac{\mathbf{D}_{1,m}}{\varepsilon_1} \\ \vdots \\ \frac{\mathbf{D}_{L,m}}{\varepsilon_L} \end{bmatrix} = \sum_{k=1}^L \varepsilon_k^{-2} \mathbf{D}_{k,m}^T \mathbf{D}_{k,m} \quad (5.27a)$$

$$\mathbf{D}_m^T \mathbf{g} = \begin{bmatrix} \frac{\mathbf{D}_{1,m}}{\varepsilon_1} \\ \vdots \\ \frac{\mathbf{D}_{L,m}}{\varepsilon_L} \end{bmatrix}^T \begin{bmatrix} \mathbf{g}_1 \\ \varepsilon_1 \\ \vdots \\ \mathbf{g}_L \\ \varepsilon_L \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{1,m}^T & \dots & \mathbf{D}_{L,m}^T \\ \varepsilon_1 & & \varepsilon_L \end{bmatrix} \begin{bmatrix} \mathbf{g}_1 \\ \varepsilon_1 \\ \vdots \\ \mathbf{g}_L \\ \varepsilon_L \end{bmatrix} = \sum_{k=1}^L \varepsilon_k^{-2} \mathbf{D}_{k,m}^T \mathbf{g}_k \quad (5.27b)$$

Vê-se, assim, que é possível substituir as operações com matrizes concatenadas pelo somatório das operações com as matrizes de cada quadro, conforme observado em:

$$\mathbf{f}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta \left(\sum_{k=1}^L (\varepsilon_k^{-2} \mathbf{D}_{k,m}^T \mathbf{g}_k) - \left(\sum_{k=1}^L (\varepsilon_k^{-2} \mathbf{D}_{k,m}^T \mathbf{D}_{k,m}) + \lambda \mathbf{C}^T \mathbf{C} \right) \mathbf{f}_m^{(n)} \right) \quad (5.28)$$

Agora, é possível comentar os detalhes computacionais do método iterativo.

1. Operador $\mathbf{C}^T \mathbf{C}$:

O operador \mathbf{C} é aplicado na imagem na forma $\mathbf{C}^T \mathbf{C}$. A implementação da operação na imagem \mathbf{f}_m é no domínio espacial, e a cada iteração na forma apresentada por:

$$(\mathbf{C}^T \mathbf{C}) \mathbf{f}_m \leftrightarrow f_m[m_1, m_2] * |C[m_1, m_2]|^2 \quad (5.29)$$

Pode-se, também, realizar a Transformada 2D de Fourier, usando a FFT (*Fast Fourier Transform*), realizar a multiplicação com o Espectro de Fourier do Filtro $\mathbf{C}^T\mathbf{C}$ e aplicar a Transformada Inversa no resultado.

2. Operador $\mathbf{D}_{k,m}$:

O operador $\mathbf{D}_{k,m}$ deve ser aplicado na forma de $\mathbf{D}_{k,m} = \mathbf{S}_k \mathbf{H}_k \mathbf{A}_{k,m}$, onde cada operador é aplicado diferentemente e em seqüência, segundo (5.30a-c). Primeiro, o operador $\mathbf{A}_{k,m}$:

$$\mathbf{f}' = \mathbf{A}_{k,m} \mathbf{f}_m \leftrightarrow f'[\mathbf{m}_k] = f_m[\mathbf{m}_k + \mathbf{v}_{k,m}[\mathbf{m}_k]], \quad (5.30a)$$

onde $\mathbf{v}_{k,m}[\mathbf{m}_k]$ são os vetores que descrevem o movimento aparente entre os *pixels* da imagem \mathbf{f}_k e da imagem \mathbf{f}_m . O vetor \mathbf{m}_k denota as coordenadas $[m_1, m_2]$ na imagem $f_k[m_1, m_2]$. Esta operação é chamada de compensação de movimento. No Capítulo 3 foi visto que as matrizes $\mathbf{A}_{k,m}$ eram criadas a partir dos vetores de movimento. Logo, pode-se concluir que realizar a operação sem criar as matrizes $\mathbf{A}_{k,m}$ é mais vantajoso.

Em seguida, é aplicada a distorção óptica \mathbf{H}_k :

$$\mathbf{f}'' = \mathbf{H}_k \mathbf{f}' \leftrightarrow f''[m_1, m_2] = f'[\mathbf{m}_1, m_2] * h_k[m_1, m_2] \quad (5.30b)$$

Esta operação tem as mesmas características de implementação que $\mathbf{C}^T\mathbf{C}$. Isso quer dizer que a operação é realizada no domínio espacial ou utilizando a FFT, mas sem criar a matriz \mathbf{H}_k .

A terceira operação é a subamostragem, \mathbf{S}_k :

$$\mathbf{f}''' = \mathbf{S}_k \mathbf{f}'' \leftrightarrow (f''[m_1, m_2] p_k[m_1, m_2])^{\downarrow \Delta m_1 \Delta m_2}, \quad (5.30c)$$

onde a função $p_k[m_1, m_2]$ é a máscara com “uns” e “zeros” conforme a grade de subamostragem. O operador $\downarrow \Delta m_1 \Delta m_2$ denota a mudança na taxa de amostragem espacial. Na prática, significa que os *pixels* de interesse são retirados da imagem $f''[m_1, m_2]$, de tamanho $M_1 \times M_2$, e transferidos para a imagem $f'''[n_1, n_2]$, de tamanho $N_1 \times N_2$.

3. Operador $\mathbf{D}_{k,m}^T$:

O operador $\mathbf{D}_{k,m}^T$ também é desmembrado em $\mathbf{D}_{k,m}^T = \mathbf{A}_{k,m}^T \mathbf{H}_k^T \mathbf{S}_k^T$. As operações são aplicadas sequencialmente, segundo (5.31a-c). Primeiramente, \mathbf{S}_k^T :

$$\mathbf{f}^{1'} = \mathbf{S}_k^T \mathbf{f}''' \leftrightarrow (f'''[n_1, n_2])^{\uparrow \Delta m_1 \Delta m_2} p_k[m_1, m_2], \quad (5.31a)$$

onde o operador $\uparrow \Delta m_1 \Delta m_2$ denota a adição de zeros entre as amostras. Pode-se notar que a multiplicação por $p_k[m_1, m_2]$ não altera em nada o resultado anterior, porque os *pixels* onde $p_k[m_1, m_2] = 0$ coincidem com os *pixels* onde $f'''[m_1, m_2] = 0$. Na prática, apenas é realizada a transferência inversa dos *pixels* da imagem $f'''[n_1, n_2]$, de tamanho $N_1 \times N_2$, para a imagem $f^{1'}[m_1, m_2]$, de tamanho $M_1 \times M_2$.

A distorção óptica, \mathbf{H}_k^T , é aplicada da mesma forma que \mathbf{H}_k . A única diferença é que o operador é o complexo conjugado:

$$\mathbf{f}^{2'} = \mathbf{H}_k^T \mathbf{f}^{1'} \leftrightarrow f^{1'}[m_1, m_2] * h_k^*[m_1, m_2], \quad (5.31b)$$

Esta operação também tem as mesmas características de implementação de $\mathbf{C}^T \mathbf{C}$ e \mathbf{H}_k .

A compensação de movimento $\mathbf{A}_{k,m}^T$ é o mesma que a compensação inversa $\mathbf{A}_{m,k}$:

$$\mathbf{f}^{3'} = \mathbf{A}_{k,m}^T \mathbf{f}^{2'} = \mathbf{A}_{m,k} \mathbf{f}^{2'} \Leftrightarrow f^{3'}[\mathbf{m}_m] = f^{2'}[\mathbf{m}_m + \mathbf{v}_{m,k}[\mathbf{m}_m]], \quad (5.31c)$$

onde \mathbf{m}_m são as coordenadas $[m_1, m_2]$, na imagem $f_m[m_1, m_2]$.

A operação $\mathbf{A}_{k,m}^T$ indica a compensação inversa de movimento e é implementada da mesma forma que $\mathbf{A}_{k,m}$, apenas com os vetores do movimento inverso.

A implementação dessas operações é mais vantajosa que o uso das matrizes principalmente porque requerem menos memória para armazenamento de variáveis. Esta seção tenta demonstrar que, se a implementação na forma matricial for inviável (e normalmente é para imagens grandes), pode-se implementar as operações em outros domínios. As necessidades da aplicação e os recursos disponíveis que determinam a forma de realização das operações do método iterativo.

5.2.3. Diferença computacional entre solução por iteração e inversão de matrizes

A diferença de custo computacional entre o cálculo da solução de forma direta, através da inversão de matrizes, e da forma iterativa é demonstrada na Tabela 5.1.

Tabela 5.1:Diferença de custo computacional

	Inversão Direta	Solução Iterativa
Forma:	$\hat{\mathbf{f}}_m = (\mathbf{B})^{-1} \mathbf{b}$	$\mathbf{f}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta(\mathbf{b} - \mathbf{B}\mathbf{f}_m^{(n)})$
Custo Computacional:	$\frac{2}{3}(M_1 M_2)^3 + 2(M_1 M_2)^2$ FLOPS	$I(3M_1 M_2 + 2(M_1 M_2)^2)$ FLOPS

onde $\mathbf{b} = \mathbf{D}_m^T \mathbf{g}$, $\mathbf{B} = (\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C})$, $M_1 M_2$ é o tamanho da matriz e I o número de iterações. O custo computacional é medido em operações de ponto flutuante (FLOPS – *Float Point Operations*). Os valores da Tabela 5.1 foram extraídos de [5,28].

Observando os valores da Tabela 5.1 pode-se notar que o custo computacional aumenta conforme o tamanho da matriz. Imagens de tamanho 100×100 , solucionadas com 35 iterações, tem aproximadamente 1% do custo computacional da solução por inversão direta.

Observa-se na Figura 5.2 a comparação entre operações necessárias para solução do sistema em relação ao tamanho da imagem. No método iterativo são consideradas 35 iterações para encontrar uma solução adequada.

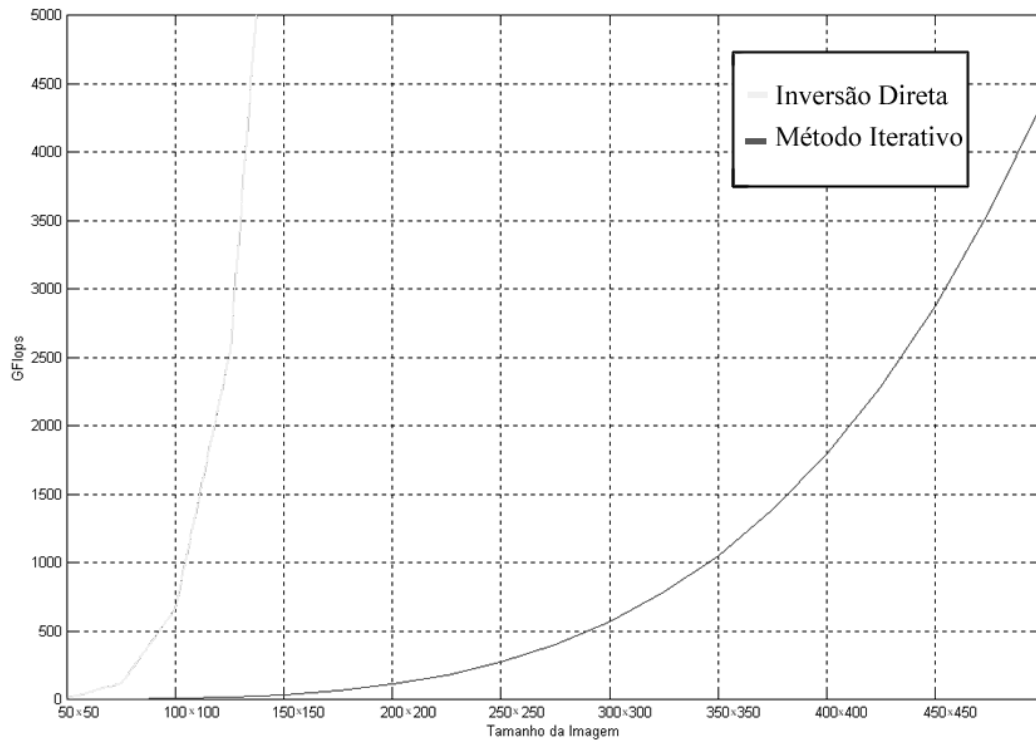


Figura 5.2: Comparação de custo computacional entre os métodos.

Conclusões do Capítulo 5

Neste capítulo, foi demonstrado que o método iterativo é mais vantajoso para a implementação da inversão da degradação que a computação da matriz inversa. Apresentaram-se duas formas de iteração bem conhecidas e estudadas na literatura. As principais propriedades do método iterativo, que levam a redução do custo computacional, foram apresentadas e explicadas.

Se o movimento entre as imagens for conhecido, têm-se todas as ferramentas para inverter a degradação e obter uma sequência de imagens de alta resolução; se não for conhecido, é preciso utilizar as técnicas para estimação do movimento. Os problemas envolvidos na estimação de movimento são apresentados no Capítulo 6.

Capítulo 6: Estimação de Movimento Entre as Imagens

Nos capítulos anteriores foram vistas as ferramentas necessárias para a estimação das imagens de alta resolução caso o movimento entre elas seja conhecido, porém existe uma grande quantidade de aplicações em que o movimento não é conhecido. Este capítulo é dedicado ao estudo dos problemas envolvidos na estimação de movimento.

No Capítulo 3 foi apresentado o movimento entre as imagens. Foi visto que, mesmo sendo o movimento desconhecido, uma série de informações sobre o movimento 3D dos objetos, precisa ser conhecida *a priori*. Assim, pode-se determinar um modelo de movimento que expresse, de forma precisa, o movimento 2D entre as imagens.

Uma vez determinado o modelo, pode-se utilizar um método de estimação apropriado, dentre os conhecidos pelas técnicas de estimação de movimento [13], para a obtenção da relação entre as imagens. Essa relação pode ser expressa pelos vetores de movimento ou parâmetros do tipo de movimento de determinada região da imagem.

Este capítulo investiga o uso da sequência de imagens de baixa resolução na estimação de movimento. Sabe-se que as relações de movimento são absolutamente válidas apenas para as imagens de alta resolução; assim, são observados erros quando as imagens de baixa resolução são utilizadas no processo de estimação. Neste capítulo, serão desenvolvidos e analisados os problemas da estimação de movimento com a sequência de baixa resolução. Serão utilizadas técnicas de combinação de intensidade (*Intensity Matching*), que é uma linha é bem conhecida e uma das mais utilizadas [10,11,14].

6.1. Estimação de Movimento Via Combinação de Intensidade

Baseando-se na relação de movimento apresentada no Capítulo 3 e reescrita em (6.1), podem-se relacionar as imagens de alta resolução por uma transformação linear.

$$\mathbf{f}_k = \mathbf{A}_{k,m} \mathbf{f}_m \quad (6.1)$$

Conforme foi mencionado nos capítulos anteriores, a transformação linear também é chamada de compensação de movimento, onde a matriz $\mathbf{A}_{k,m}$ indica a transformação que irá retornar o movimento sofrido pela imagem \mathbf{f}_m , para que se torne igual à imagem \mathbf{f}_k .

Como não se conhece o movimento, não se conhece a matriz $\mathbf{A}_{k,m}$; sabe-se apenas que ela expressa o movimento determinado pelo modelo de movimento escolhido. Dessa forma, pode-se dizer que a matriz $\mathbf{A}_{k,m}$ é uma função dependente dos parâmetros do modelo de movimento, e são os parâmetros do modelo de movimento que realmente precisam ser estimados. A Expressão (6.2)

indica que a matriz de movimento realiza uma transformação descrita pelo modelo de movimento e seus parâmetros $[\theta_1, \dots, \theta_N]$:

$$\hat{\mathbf{A}}_{k,m} = \mathbf{A}_{k,m}(\boldsymbol{\theta}), \quad (6.2)$$

onde $\boldsymbol{\theta} = [\theta_1 \dots \theta_N]$ é o vetor com o conjunto de parâmetros.

Observando a Equação (6.1), pode-se notar que, se a matriz de compensação de movimento não estiver correta, tem-se um erro, expresso por:

$$\mathbf{e}_{k,m} = \mathbf{f}_k - \hat{\mathbf{A}}_{k,m} \mathbf{f}_m, \quad (6.3)$$

onde $\mathbf{e}_{k,m}$ é um vetor que expressa o erro entre a imagem \mathbf{f}_k e a imagem \mathbf{f}_m com movimento compensado através da matriz de movimento estimada $\hat{\mathbf{A}}_{k,m}$. Como a matriz é função dos parâmetros do modelo de movimento, variando os parâmetros, tem-se uma matriz diferente e, conseqüentemente, um vetor de erro diferente. Esta relação é descrita por:

$$\mathbf{e}_{k,m}(\boldsymbol{\theta}) = \mathbf{f}_k - \mathbf{A}_{k,m}(\boldsymbol{\theta}) \mathbf{f}_m, \quad (6.4)$$

onde $\mathbf{e}_{k,m}(\boldsymbol{\theta})$ é uma função vetor dependente dos parâmetros do modelo de movimento. Essa função é N dimensional, pois têm-se N parâmetros no modelo de movimento e, em cada ponto nas N dimensões, existe um vetor de tamanho $M_1 M_2$.

Embora as equações tenham sido apresentadas na forma vetorial, a Equação (6.4) pode ser apresentada na forma tradicional, como função escalar bidimensional, segundo (6.5). Essa operação é conhecida como diferença entre quadros deslocados (DFD – *Displaced Frame Difference*).

$$DFD_{k,m}[\mathbf{m}_k] = e_{k,m}[\mathbf{m}_k] = f_k[\mathbf{m}_k] - f_m[\mathbf{m}_k + \mathbf{v}_{k,m}[\mathbf{m}_k]], \quad (6.5)$$

onde \mathbf{m}_k são as coordenadas $[m_1, m_2]$ na grade HR, indicando que estão nas condições espaço-temporais da projeção k . Os vetores $\mathbf{v}_{k,m}[\mathbf{m}_k]$ são os vetores de movimento utilizados para compensar a imagem $f_m[m_1, m_2]$.

Em virtude da complexidade e do tamanho dos dados resultantes das Equações (6.3-6.5), é necessário buscar formas para descrever melhor tais relações, tornando-as próprias para a busca dos parâmetros corretos através de otimização. Uma forma bem conhecida é o uso da norma do vetor $\mathbf{e}_{k,m}$, o que será visto a seguir.

6.1.1. Minimização da norma quadrática

Uma forma de se minimizar as componentes do vetor de erro $\mathbf{e}_{k,m}$ na busca do movimento correto é minimizar a norma do vetor $\mathbf{e}_{k,m}$, segundo:

$$\hat{\mathbf{A}}_{k,m} = \arg \min_{\mathbf{A}_{k,m}} \|\mathbf{e}_{k,m}\|_2^2 = \arg \min_{\mathbf{A}_{k,m}} \|\mathbf{f}_k - \mathbf{A}_{k,m} \mathbf{f}_m\|_2^2 \quad (6.6a)$$

A equação pode ser expressa em função dos parâmetros da matriz de movimento, segundo:

$$\hat{\theta} = \arg \min_{\theta} \left\| \mathbf{f}_k - \mathbf{A}_{k,m}(\theta) \mathbf{f}_m \right\|_2^2 \quad (6.6b)$$

Observando as Equações (6.6a) e (6.6b), nota-se que $\|\mathbf{e}_{k,m}\|_2^2$ é uma função de objetivo escalar e N dimensional e, possivelmente, não será convexa.

A norma quadrática pode ser representada por:

$$\left\| \mathbf{e}_{k,m} \right\|_2^2 = (\mathbf{f}_k^T \mathbf{f}_k + \mathbf{f}_m^T \mathbf{f}_m) - (\mathbf{f}_k^T \mathbf{A}_{k,m} \mathbf{f}_m + \mathbf{f}_m^T \mathbf{A}_{m,k} \mathbf{f}_k), \quad (6.7)$$

onde os termos $\mathbf{f}_k^T \mathbf{A}_{k,m} \mathbf{f}_m$ e $\mathbf{f}_m^T \mathbf{A}_{m,k} \mathbf{f}_k$ podem ser observados de forma similar a uma função *correlação cruzada*. Esta correlação pode ser chamada de *correlação de movimento*, pois é a similaridade entre as imagens \mathbf{f}_k e \mathbf{f}_m que está sendo medida, conforme o movimento entre elas muda. A forma da função é extremamente dependente das imagens e das características 3D da cena em que elas foram formadas.

Conclui-se que a forma da função objetivo depende da forma da função correlação de movimento, ou seja, se a correlação for côncava, a função de objetivo será convexa. Espera-se que a função objetivo tenha características que permitam a otimização [5,7,13], para que o movimento correto possa ser estimado.

6.2. Estimação de Movimento de Imagens de Alta Resolução a partir de Imagens de Baixa Resolução

Um dos maiores problemas da restauração da resolução de sequência de imagens é a estimação do movimento a partir das imagens degradadas. Este assunto ainda é tópico de intensas pesquisas [10,11,14] contudo ainda não se tem uma definição sobre a melhor forma de estimar o movimento a partir de imagens degradadas.

Como apenas as imagens degradadas, \mathbf{g}_k e \mathbf{g}_m , estão disponíveis e não se conhece o movimento aparente entre elas, é preciso utilizar algum método de restauração de degradação baseado em uma única imagem para produzir estimativas de \mathbf{f}_k e \mathbf{f}_m . Como as degradações de única imagem, apresentadas em (6.9), são matrizes subdeterminadas, as imagens degradadas sofrem perda de informação que não pode ser recuperada através da restauração de única imagem. Assim, qualquer estimação ou restauração de única imagem irá conter erros. Observe-se a Equação (6.8).

$$\begin{aligned} \hat{\mathbf{f}}_k &= \mathbf{B}_k \mathbf{g}_k = \mathbf{f}_k + \mathbf{e}_k \\ \hat{\mathbf{f}}_m &= \mathbf{B}_m \mathbf{g}_m = \mathbf{f}_m + \mathbf{e}_m, \end{aligned} \quad (6.8)$$

onde a equação de degradação de única imagem é apresentada em (6.9).

$$\begin{aligned} \mathbf{g}_k &= \mathbf{S}_k \mathbf{H}_k \mathbf{f}_k + \boldsymbol{\eta}_k = \mathbf{D}_k \mathbf{f}_k + \boldsymbol{\eta}_k \\ \mathbf{g}_m &= \mathbf{S}_m \mathbf{H}_m \mathbf{f}_m + \boldsymbol{\eta}_m = \mathbf{D}_m \mathbf{f}_m + \boldsymbol{\eta}_m \end{aligned} \quad (6.9)$$

Assim, a estimativa de alta resolução tem de corrigir a degradação usando uma única imagem. As matrizes \mathbf{B}_k e \mathbf{B}_m são filtros restauradores que tentam corrigir as degradações causadas a \mathbf{g}_k e \mathbf{g}_m , respectivamente. Em alguns casos, esta realização tem de ser iterativa, mas, em outros, filtros espaço-invariantes podem ser utilizados nas inversões de degradação de única imagem. Como a degradação não pode ser totalmente invertida, existirá um erro de estimação de cada imagem, o qual pode ser expresso por:

$$\begin{aligned}
 \mathbf{e}_k &= \mathbf{B}_k \mathbf{g}_k - \mathbf{f}_k & \mathbf{e}_m &= \mathbf{B}_m \mathbf{g}_m - \mathbf{f}_m \\
 \mathbf{e}_k &= \mathbf{B}_k (\mathbf{D}_k \mathbf{f}_k + \boldsymbol{\eta}_k) - \mathbf{f}_k & \mathbf{e}_m &= \mathbf{B}_m (\mathbf{D}_m \mathbf{f}_m + \boldsymbol{\eta}_m) - \mathbf{f}_m \\
 \mathbf{e}_k &= (\mathbf{B}_k \mathbf{D}_k - \mathbf{I}) \mathbf{f}_k + \mathbf{B}_k \boldsymbol{\eta}_k & \mathbf{e}_m &= (\mathbf{B}_m \mathbf{D}_m - \mathbf{I}) \mathbf{f}_m + \mathbf{B}_m \boldsymbol{\eta}_m \\
 \mathbf{e}_k &= \mathbf{E}_k \mathbf{f}_k + \mathbf{B}_k \boldsymbol{\eta}_k & \mathbf{e}_m &= \mathbf{E}_m \mathbf{f}_m + \mathbf{B}_m \boldsymbol{\eta}_m
 \end{aligned} \tag{6.10}$$

Grande parte dos autores [10,11,14] sugere o uso de uma estimativa inicial das imagens de alta resolução para utilização na técnica de estimação de movimento. Alguns trabalhos [10,11] apresentam a interpolação bilinear como uma estimativa inicial aceitável. Nesta condição, a matriz \mathbf{B}_k é igual a \mathbf{B}_m , e ambas realizam uma operação de interpolação bilinear.

Seja qual for a estimativa inicial, ao ser usada na procura do movimento, observa-se um efeito na norma do erro, dado por:

$$\hat{\mathbf{A}}_{k,m} = \arg \min_{\mathbf{A}_{k,m}} \|\hat{\mathbf{e}}_{k,m}\|_2^2 = \arg \min_{\mathbf{A}_{k,m}} \|\hat{\mathbf{f}}_k - \mathbf{A}_{k,m} \hat{\mathbf{f}}_m\|_2^2 = \arg \min_{\mathbf{A}_{k,m}} \|\mathbf{f}_k + \mathbf{e}_k - \mathbf{A}_{k,m} (\mathbf{f}_m + \mathbf{e}_m)\|_2^2 \tag{6.11a}$$

Agora, a norma do erro envolve não apenas o erro de movimento entre as imagens, mas, também, o erro de estimação das imagens de alta resolução. Observe-se em (6.11b):

$$\hat{\mathbf{A}}_{k,m} = \arg \min_{\mathbf{A}_{k,m}} \|\hat{\mathbf{e}}_{k,m}\|_2^2 = \arg \min_{\mathbf{A}_{k,m}} \|(\mathbf{f}_k - \mathbf{A}_{k,m} \mathbf{f}_m) + (\mathbf{e}_k - \mathbf{A}_{k,m} \mathbf{e}_m)\|_2^2 \tag{6.11b}$$

Se for considerado que nas Equações (6.6a-b) o ponto ótimo pode ser encontrado e tem norma zero, em (6.11a-b) isso não pode mais ser garantido, principalmente porque a forma da função objetivo é afetada e a norma mínima de $\hat{\mathbf{e}}_{k,m}$ pode não ser mais zero. Portanto, a estimativa inicial das imagens de baixa resolução deve ser escolhida de forma a afetar o mínimo possível a forma da função. As interpolações bilinear e bicúbica são aceitáveis na maioria das degradações \mathbf{D}_k e \mathbf{D}_m , pois geram estimativas suaves e, em alguns casos, até reduzem as perturbações da função objetivo. A Figura 6.1 demonstra a diferença na forma da função objetivo usando as imagens originais e as estimadas por interpolação bicúbica. Esta figura mostra o efeito em uma das dimensões.

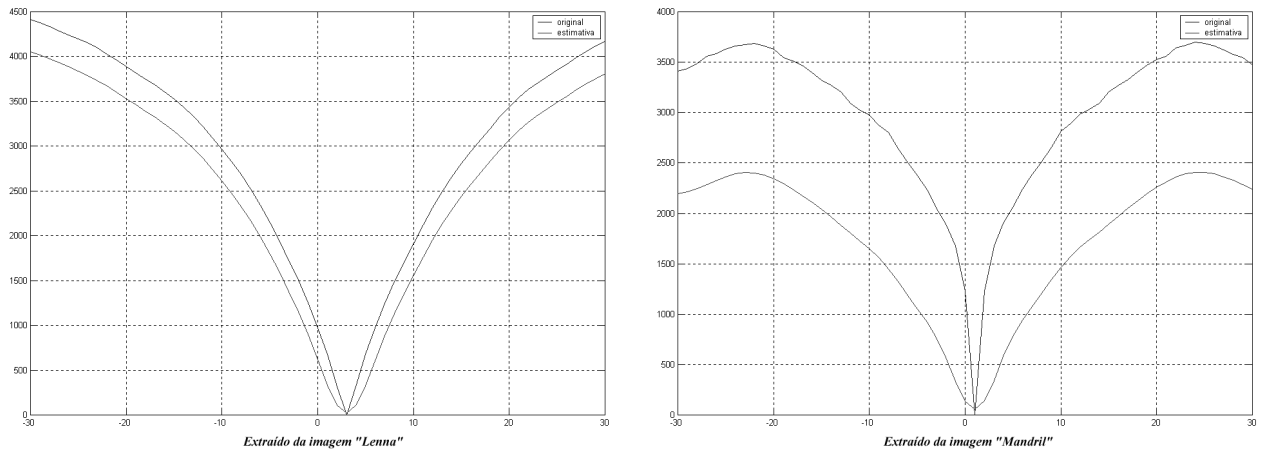


Figura 6.1: Corte unidimensional de funções objetivo usando as imagens originais e as imagens estimadas com interpolação bicúbica.

Os exemplos ilustrados na Figura 6.1 mostram que a forma da função objetivo depende da imagem e da correlação de movimento. Nos dois casos, a curva da função objetivo têm o mínimo no mesmo ponto (o corte foi realizado no ponto de mínimo nas outras dimensões). Pode-se observar que o erro mínimo não será zero quando as imagens estimadas são utilizadas.

A Equação (6.12) apresenta o erro mínimo no caso do uso das imagens estimadas:

$$\min \|\hat{\mathbf{e}}_{k,m}\|_2^2 = \|\mathbf{e}_k - \mathbf{A}_{k,m} \mathbf{e}_m\|_2^2 = \|(\mathbf{E}_k \mathbf{f}_k - \mathbf{A}_{k,m} \mathbf{E}_m \mathbf{f}_m) + (\mathbf{B}_k \boldsymbol{\eta}_k - \mathbf{A}_{k,m} \mathbf{B}_m \boldsymbol{\eta}_m)\|_2^2 \quad (6.12)$$

Esse erro mínimo é proveniente da diferença entre a informação dos detalhes perdidos na degradação, mais a diferença entre o ruído filtrado pelo filtro de restauração de única imagem. Esta informação será usada na técnica de estimação da região de *outliers*. Mesmo assim, não há garantias de que o mínimo seja encontrado, ou de que o mínimo global das Equações (6.11a-b) seja o movimento correto. Portanto, não se pode confiar absolutamente no movimento estimado.

Como o movimento estimado pode não ser correto, as equações dos sistemas lineares fornecidos por cada quadro conterão um erro maior que o ruído, o qual deve ser considerado na restauração da resolução. Na seção seguinte será explicado como o erro é considerado.

6.2.1. Definindo ponderação do sistema linear de cada imagem de acordo com o erro de estimação de movimento

No Capítulo 4 foi definido que cada imagem, em conjunto com sua degradação, forma um sistema linear. Foi definido também que, se o movimento aparente entre as imagens for conhecido, é possível concatenar os sistemas de cada quadro para formar um sistema com mais equações, melhorando a qualidade da resposta.

Ainda no Capítulo 4, mostrou-se que, através do uso de conjuntos teóricos, pode-se preservar a individualidade de cada imagem da seqüência. Isso implica ponderar cada conjunto de equações adequadamente, segundo a norma do ruído ou algum erro de estimação.

Pode-se demonstrar que o movimento estimado pode conter erros, os quais devem ser considerados. Se for substituída a Equação (6.3) em (6.9), é possível observar o erro de estimação, conforme:

$$\mathbf{g}_k = \mathbf{D}_k (\hat{\mathbf{A}}_{k,m} \mathbf{f}_m + \mathbf{e}_{k,m}) + \boldsymbol{\eta}_k \quad (6.13)$$

Logo, tem-se que a equação que representa o conjunto de respostas abrange um conjunto maior de soluções, como:

$$\|\mathbf{g}_k - \mathbf{D}_k \hat{\mathbf{A}}_{k,m} \mathbf{f}_m\|_2 = \|\mathbf{D}_k \mathbf{e}_{k,m} + \boldsymbol{\eta}_k\|_2 \quad (6.14)$$

Portanto, é preciso escolher um fator de ponderação, ϵ_k , que leve em conta a norma do novo erro. Anteriormente, ϵ_k era escolhido de forma a satisfazer o ruído, $\epsilon_k \geq \|\boldsymbol{\eta}_k\|_2$; agora, deve satisfazer $\epsilon_k \geq \|\mathbf{D}_k \mathbf{e}_{k,m} + \boldsymbol{\eta}_k\|_2$. Como o vetor de ruído não é conhecido (apenas a norma é conhecida), não se pode determinar um valor exato para ϵ_k . Assim, normalmente, trabalha-se com o pior caso, segundo:

$$\epsilon_k \geq \|\mathbf{D}_k \mathbf{e}_{k,m}\|_2 + \|\boldsymbol{\eta}_k\|_2 \quad (6.15)$$

No trabalho [10], o fator ϵ_k é escolhido de acordo com (6.16), por já considerar um valor aceitável englobando ruído e erros de estimação:

$$\epsilon_k \geq \|\hat{\mathbf{e}}_{k,m}\|_2 \quad (6.16)$$

Dado que não se pode obter $\mathbf{e}_{k,m}$, apenas $\hat{\mathbf{e}}_{k,m}$, por causa da não-disponibilidade das imagens de alta resolução, a Equação (6.16) pode ser uma base para a escolha da ponderação.

6.3. Estimando Movimento com *Inliers* e *Outliers*

Na Seção 3.3 foi apresentado o tratamento do movimento entre imagens, quando apenas parte delas pode ser relacionada. Demonstrou-se que as regiões consideradas *inliers* podem ser relacionadas por movimento. Quanto às regiões consideradas *outliers*, não podem ser relacionadas e devem ser desconsideradas na restauração da resolução, o que torna a equação de movimento em (6.1) imprópria. Portanto, a Equação (6.17) é utilizada:

$$\mathbf{f}_k^I = \mathbf{A}_{k,m}^I \mathbf{f}_m \quad (6.17)$$

Em virtude da modificação na equação de movimento, tem-se de modificar todo o equacionamento, até as funções objetivo de estimação de movimento. Assim, a matriz de movimento estimada agora é representada por:

$$\hat{\mathbf{A}}_{k,m}^I = \hat{\mathbf{I}}_k \hat{\mathbf{A}}_{k,m} = \hat{\mathbf{I}}_k \mathbf{A}_{k,m}(\hat{\boldsymbol{\theta}}), \quad (6.18)$$

onde é preciso estimar tanto o movimento quanto a região de *inliers*.

Logo, o erro devido ao uso do movimento estimado é descrito como:

$$\mathbf{e}_{k,m}^I = \mathbf{f}_k^I - \hat{\mathbf{A}}_{k,m}^I \mathbf{f}_m = \hat{\mathbf{I}}_k (\mathbf{f}_k - \hat{\mathbf{A}}_{k,m} \mathbf{f}_m) = \hat{\mathbf{I}}_k \mathbf{e}_{k,m} \quad (6.19)$$

Assim, pode-se seguir modificando as equações de minimização.

6.3.1. Minimização da norma do erro com *inliers* e *outliers*

A mudança da relação de movimento entre as imagens devido aos *inliers* e *outliers* implica também a mudança das equações de minimização. Portanto, a função objetivo deve ser escrita como:

$$\hat{\mathbf{A}}_{k,m}^I = \arg \min_{\mathbf{I}_k, \mathbf{A}_{k,m}} \|\mathbf{e}_{k,m}^I\|_2^2 = \arg \min_{\mathbf{I}_k, \mathbf{A}_{k,m}} \|\mathbf{I}_k (\mathbf{f}_k - \mathbf{A}_{k,m} \mathbf{f}_m)\|_2^2 \quad (6.20)$$

A norma quadrática do erro pode ser expressa como:

$$\|\mathbf{e}_{k,m}^I\|_2^2 = (\mathbf{f}_k^T \mathbf{I}_k \mathbf{f}_k + \mathbf{f}_m^T \mathbf{A}_{m,k} \mathbf{I}_k \mathbf{A}_{k,m} \mathbf{f}_m) - (\mathbf{f}_k^T \mathbf{I}_k \mathbf{A}_{k,m} \mathbf{f}_m + \mathbf{f}_m^T \mathbf{A}_{m,k} \mathbf{I}_k \mathbf{f}_k) \quad (6.21)$$

Analizando a Equação (6.21), nota-se que, além de achar o movimento, é necessário achar a melhor matriz para os *inliers*, pois estes também são desconhecidos. Porém, a Equação (6.21) possui uma solução mínima errônea que deve ser levada em conta. O problema ocorre porque, ao considerar mais *pixels outliers* e menos *pixels inliers*, a matriz \mathbf{I}_k passa a conter mais zeros, reduzindo a norma do erro. Logo, a otimização irá achar o mínimo considerando todos os *pixels outliers*.

Para corrigir esse problema, é preciso utilizar a norma quadrática média (*average power*), ou seja, a norma do erro proporcional ao número de *pixels inliers*. Este equacionamento é apresentado em (6.22):

$$\hat{\mathbf{A}}_{k,m}^I = \arg \min_{\mathbf{I}_k, \mathbf{A}_{k,m}} \frac{\|\mathbf{e}_{k,m}^I\|_2^2}{N_I} = \arg \min_{\mathbf{I}_k, \mathbf{A}_{k,m}} \frac{\|\mathbf{I}_k (\mathbf{f}_k - \mathbf{A}_{k,m} \mathbf{f}_m)\|_2^2}{N_I}, \quad (6.22)$$

onde N_I é o número de *pixel inliers* na imagem \mathbf{f}_k ou na imagem \mathbf{f}_m .

6.4. Estimação de Movimento e de *Inliers* e *Outliers* a Partir de Imagens de Baixa Resolução

Nesta seção, o caso que envolve os problemas práticos é demonstrado. Além da necessidade de estimar o movimento usando imagens obtidas da restauração de única imagem, é necessário identificar as regiões de *inliers* e *outliers*.

As mesmas imagens restauradas apresentadas pela Equação (6.8) serão utilizadas na função objetivo (6.22) e o resultado é apresentado em (6.23):

$$\hat{\mathbf{A}}_{k,m}^I = \arg \min_{\mathbf{I}_k, \mathbf{A}_{k,m}} \frac{\|\hat{\mathbf{e}}_{k,m}^I\|_2^2}{N_I} = \arg \min_{\mathbf{I}_k, \mathbf{A}_{k,m}} \frac{\|\mathbf{I}_k(\hat{\mathbf{f}}_k - \mathbf{A}_{k,m}\hat{\mathbf{f}}_m)\|_2^2}{N_I} \quad (6.23)$$

Esta é a forma como a estimação de movimento acontece nos trabalhos [10,11,14] e como será realizado nesse.

Devido à mudança na quantidade de *pixels inliers* em cada imagem, a escolha de ϵ_k também muda. Em vez de (6.16), é usado (6.24):

$$\epsilon_k^2 \geq \frac{\|\hat{\mathbf{e}}_{k,m}^I\|_2^2}{N_I} \quad (6.24)$$

Por causa dessas mudanças nos fatores ϵ_k , onde $k=1\dots L$, o fator E deve ser escolhido como energia média, ou seja, $E^2 \geq \|Cf_m\|^2/M_1M_2$.

Na próxima seção, será apresentada a forma mais simples que é utilizada na detecção de *inliers* e *outliers*.

6.4.1. Separando *inliers* e vetores bem estimados dos *outliers* e vetores mal estimados

Na maioria das situações práticas, as regiões de *inliers* e *outliers* são desconhecida; muitas vezes não se sabe se elas, de fato, existem no movimento (com exceção das bordas). Portanto, os procedimentos de estimação normalmente não buscam por estas regiões até que o erro esteja próximo do mínimo. Essa forma de estimação é um pouco arriscada se a maior parte da imagem for composta de *outliers*, mas é bem segura se for composta por *inliers*.

Como se está lidando com algoritmos de combinação de intensidade, espera-se que, próximo ao mínimo, o erro *pixel a pixel* seja pequeno. Dessa forma, todo e qualquer erro acima de uma determinada amplitude será decorrente da comparação entre regiões diferentes. Isso porque, ou o vetor foi mal-estimado (ainda não foi encontrado o movimento aparente correto), ou as regiões comparadas são *outliers*. Baseando-se nisso, os algoritmos normalmente iniciam um processo de refinamento de estimação ao se aproximarem do mínimo, excluindo regiões com amplitudes de erro muito grandes, como em [22].

Em algoritmos mais simples e menos robustos, a região de *inliers* e *outliers* é determinada após o mínimo ter sido encontrado. Estas técnicas são baseadas no fato de que a presença de *outliers* em pouca quantidade não prejudica a estimação de movimento, e os *inliers* e *outliers* podem, portanto, ser separados em um processamento posterior, como em [11,14].

Essas técnicas baseiam-se no uso de um processamento de detecção e eliminação de *outliers* e vetores mal-estimados. Esse processamento detecta os *outliers* e vetores mal-estimados através da

amplitude de cada *pixel* no erro de estimação $\mathbf{e}_{k,m}$, ou $DFD_{k,m}[\mathbf{m}_k]$. Assim, se o valor absoluto do erro em determinado *pixel* for maior que um limiar T , determinado segundo as estatísticas das imagens ou do erro, esse é considerado *outlier* ou possui vetor mal-estimado. Em qualquer dos casos, ele não deverá ser usado na restauração da resolução. A etapa de detecção e eliminação de *outliers* normalmente é realizada durante ou após a estimação de movimento e antes da restauração da resolução.

Em [14] T é determinado por:

$$T = \mu_{DFD} + 2\sigma_{DFD}, \quad (6.25)$$

onde μ_{DFD} é a média de $|DFD_{k,m}[\mathbf{m}_k]|$ e σ_{DFD} é o desvio-padrão. Esta técnica parece achar um bom valor para o limiar, porém o trabalho não apresenta detalhes de como chegou a essas equações nem cita referências sobre isso.

A análise apresentada neste capítulo não permite determinar um valor de T ótimo; apenas permite concluir que o valor correto deve preservar o erro mínimo apresentado na Equação (6.12) e eliminar os erros causados pelos *outliers* e vetores mal-estimados. Isso porque o erro mínimo, de certa forma, está relacionado com a contribuição que a imagem irá trazer à restauração da resolução.

Conclusões do Capítulo 6

Neste capítulo, observaram-se os problemas nas técnicas de estimação de movimento a partir das imagens degradadas. Foi mostrado que é preciso utilizar uma estimativa da imagem de alta resolução baseado na restauração de única imagem. Por causa da degradação, a estimativa de única imagem conterá erros, os quais prejudicam a estimação de movimento.

Além dos erros na estimativa, existe também o problema da presença de *inliers* e *outliers*. Em razão da existência dessas regiões, deve-se utilizar um método de estimação que identifique e mantenha apenas os *inliers* na relação de movimento entre as imagens.

Ficou demonstrado que o ponto mínimo nas equações de estimação de movimento pode conter um erro remanescente, o qual pode ser usado para ponderar os quadros na concatenação de formação do sistema linear com sequência de imagens. A ponderação, ϵ_k , que foi vista no Capítulo 4, serve para reduzir a influência das imagens da sequência com muito ruído, ou com movimento mal-estimado. A estimação de movimento entre imagens degradadas, e ainda com *inliers* e *outliers*, é uma área aberta a pesquisas onde muito ainda pode ser feito.

Capítulo 7: Solução Iterativa de Sequência de Imagens com *Inliers*

Nos Capítulos 4 e 5, foi vista a solução iterativa de sequência de imagens caso o movimento seja conhecido. No Capítulo 6 foram apresentados detalhes sobre a estimação de movimento caso o movimento não seja conhecido. Também foi demonstrada a necessidade de detecção e eliminação de *outliers* e vetores mal-estimados na sequência de imagens, pois estes prejudicam a qualidade da imagem estimada [1]. Nesses casos, o movimento não relaciona as imagens inteiramente e as equações têm que ser modificadas. Isso leva a uma eventual mudança no modelo de degradação, para englobar a presença de *inliers* e *outliers*. Este capítulo dedica-se a apresentar as mudanças no método iterativo de sequência de imagens devido às presenças de *inliers* e *outliers* no movimento entre as imagens.

7.1. Modelo de Degradação Considerando *Inliers* e *Outliers*

No Capítulo 2, foi apresentado o modelo de degradação na Equação (2.11), reescrito aqui na Equação (7.1).

$$\mathbf{g}_k = \mathbf{S}_k \mathbf{H}_k \mathbf{A}_{k,m} \mathbf{f}_m + \boldsymbol{\eta}_k \quad (7.1)$$

Conforme foi apresentado nos Capítulos 3 e 6, a relação de movimento não é mais válida e deve ser modificada para:

$$\mathbf{f}_k = \mathbf{f}_k^I + \mathbf{f}_k^O = \mathbf{I}_k \mathbf{A}_{k,m} \mathbf{f}_m + \mathbf{O}_k \mathbf{f}_k, \quad (7.2a)$$

onde os *inliers* da imagem \mathbf{f}_k , chamados \mathbf{f}_k^I , podem ser relacionados com os *inliers* na imagem \mathbf{f}_m . Já os *outliers* da imagem \mathbf{f}_k , chamados \mathbf{f}_k^O , só podem ser relacionados com eles mesmos.

$$\begin{aligned} \mathbf{g}_k &= \mathbf{S}_k \mathbf{H}_k (\mathbf{I}_k \mathbf{A}_{k,m} \mathbf{f}_m + \mathbf{O}_k \mathbf{f}_k) + \boldsymbol{\eta}_k \\ \mathbf{g}_k - (\mathbf{S}_k \mathbf{H}_k \mathbf{O}_k \mathbf{f}_k + \boldsymbol{\eta}_k^O) &= \mathbf{S}_k \mathbf{H}_k \mathbf{I}_k \mathbf{A}_{k,m} \mathbf{f}_m + \boldsymbol{\eta}_k^I \\ \mathbf{g}_k^I &= \mathbf{S}_k \mathbf{H}_k \mathbf{A}_{k,m}^I \mathbf{f}_m + \boldsymbol{\eta}_k^I \end{aligned} \quad (7.2b)$$

Na Equação (7.2b), \mathbf{g}_k^I é a imagem de baixa resolução composta apenas pelos *pixels inliers* da imagem k . O ruído $\boldsymbol{\eta}_k^I$ tem as mesmas propriedades de $\boldsymbol{\eta}_k$; a diferença é que irá atuar apenas na região de *inliers* e terá uma norma menor.

Como, inicialmente, não se conhece a imagem \mathbf{f}_k , para que se possa produzir \mathbf{g}_k^I , \mathbf{g}_k^I é aproximada, segundo:

$$\mathbf{g}_k^I \cong \mathbf{i}_k^\downarrow \bullet \mathbf{g}_k = \mathbf{I}_k^\downarrow \mathbf{g}_k, \quad (7.3)$$

onde \mathbf{I}_k^\downarrow é uma matriz diagonal, de tamanho $N_1 N_2 \times N_1 N_2$, cuja diagonal principal é composta pelo vetor \mathbf{i}_k^\downarrow . O vetor \mathbf{i}_k^\downarrow corresponde à máscara de *inliers* de baixa resolução e é produzido segundo:

$$\mathbf{i}_k^\downarrow = \mathbf{S}_k \mathbf{H}_k \mathbf{i}_k, \quad (7.4)$$

onde \mathbf{i}_k é um vetor que compõe a máscara de *inliers* de alta resolução da imagem \mathbf{f}_k .

Dessa forma, tem-se um novo modelo de degradação, considerando apenas o movimento dos *inliers*:

$$\mathbf{g}_k^I = \mathbf{D}_{k,m}^I \mathbf{f}_m + \boldsymbol{\eta}_k^I, \quad (7.5)$$

onde a matriz $\mathbf{D}_{m,k}^I = \mathbf{S}_k \mathbf{H}_k \mathbf{I}_k \mathbf{A}_{k,m}$.

Essa alteração pode ser diretamente incorporada no equacionamento da solução iterativa [15,16]. A seção seguinte apresenta essas modificações.

7.2. Mudança no Equacionamento da Solução Iterativa

No Capítulo 4, foi apresentada a solução de Mínimos Quadrados Regularizados utilizando Conjuntos Teóricos ponderados [16]. Por causa da mudança no modelo de degradação, a função objetivo deve ser mudada. Na Equação (7.6), é apresentado o novo conjunto de equações que englobam a solução:

$$\text{satisfazer:} \begin{cases} \|\mathbf{C}\mathbf{f}_m\|_2 \leq E \\ \|\mathbf{g}_1^I - \mathbf{D}_{1,m}^I \mathbf{f}_m\|_2 \leq \varepsilon_1 \\ \vdots \\ \|\mathbf{g}_L^I - \mathbf{D}_{L,m}^I \mathbf{f}_m\|_2 \leq \varepsilon_L \end{cases}, \quad (7.6)$$

onde as relações $\|\mathbf{g}_k^I - \mathbf{D}_{k,m}^I \mathbf{f}_m\|_2 \leq \varepsilon_k$ descrevem o conjunto teórico de soluções possíveis, proporcionadas por cada quadro da seqüência, considerando apenas os *inliers*. É interessante notar que apenas na imagem \mathbf{f}_m o modelo de degradação não muda, pois não há movimento. Neste caso, as equações são iguais $\|\mathbf{g}_m^I - \mathbf{D}_m^I \mathbf{f}_m\|_2 = \|\mathbf{g}_m - \mathbf{D}_m \mathbf{f}_m\|_2$.

O novo conjunto de equações leva à função objetivo:

$$\hat{\mathbf{f}}_m^{RLS} = \arg \min_f \frac{\kappa \|\mathbf{C}\mathbf{f}_m\|_2^2}{E^2} + \frac{\|\mathbf{g}_1^I - \mathbf{D}_{1,m}^I \mathbf{f}_m\|_2^2}{\varepsilon_1^2} + \dots + \frac{\|\mathbf{g}_L^I - \mathbf{D}_{L,m}^I \mathbf{f}_m\|_2^2}{\varepsilon_L^2}, \quad (7.7)$$

cujas solução é dada por (7.8).

$$\hat{\mathbf{f}}_m^{RLS} = \left(\sum_{k=1}^L \varepsilon_k^{-2} (\mathbf{D}_{k,m}^I)^T \mathbf{D}_{k,m}^I + \lambda \mathbf{C}^T \mathbf{C} \right)^{-1} \sum_{k=1}^L \varepsilon_k^{-2} (\mathbf{D}_{k,m}^I)^T \mathbf{g}_k^I, \quad \text{onde } \lambda = \frac{\kappa}{E^2} \quad (7.8)$$

Seguindo os mesmos passos do desenvolvimento apresentados no Capítulo 4, tem-se que a solução para a inversão de (7.8) deve ser encontrada de forma iterativa. Mas, antes, será demonstrada a concatenação do conjunto de equações em (7.6). Conforme a teoria de conjuntos teóricos ponderados, a concatenação pode ser expressa como:

$$\begin{bmatrix} \mathbf{g}_1^I \\ \varepsilon_1 \\ \vdots \\ \mathbf{g}_L^I \\ \varepsilon_L \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{1,m}^I \\ \varepsilon_1 \\ \vdots \\ \mathbf{D}_{L,m}^I \\ \varepsilon_L \end{bmatrix} \mathbf{f}_m + \begin{bmatrix} \boldsymbol{\eta}_1^I \\ \varepsilon_1 \\ \vdots \\ \boldsymbol{\eta}_L^I \\ \varepsilon_1 \end{bmatrix}, \text{ ou } \mathbf{g}^I = \mathbf{D}_m^I \mathbf{f}_m + \boldsymbol{\eta}^I, \quad (7.9)$$

onde o vetor \mathbf{g}^I representa os *inliers* de cada imagem concatenados; a matriz \mathbf{D}_m^I representa as matrizes $\mathbf{D}_{k,m}^I$ concatenadas. O vetor $\boldsymbol{\eta}^I$ não é conhecido, assim como $\boldsymbol{\eta}$; apenas a norma $\|\boldsymbol{\eta}^I\|_2$ é conhecida e é menor do que a norma $\|\boldsymbol{\eta}\|_2$, levando em conta a quantidade de *pixels outliers* que foram eliminados.

A solução da equação de minimização, representada através das matrizes e vetores concatenados, está expressa abaixo:

$$\hat{\mathbf{f}}_m^{RLS} = ((\mathbf{D}_m^I)^T \mathbf{D}_m^I + \lambda \mathbf{C}^T \mathbf{C})^{-1} (\mathbf{D}_m^I)^T \mathbf{g}^I \quad (7.10)$$

Então, a solução iterativa pode ser reformulada para uso das equações relacionadas apenas aos *pixels inliers* das imagens. A Equação (7.11) apresenta este resultado.

$$\mathbf{f}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta \left((\mathbf{D}_m^I)^T \mathbf{g}^I - ((\mathbf{D}_m^I)^T \mathbf{D}_m^I + \lambda \mathbf{C}^T \mathbf{C}) \mathbf{f}_m^{(n)} \right) \quad (7.11)$$

Agora a equação é própria para uso com seqüência de imagens que contenham *pixels inliers* e *outliers* presentes em sua relação de movimento.

7.2.1. Escolha prática dos fatores de regularização

Nas situações práticas, as escolhas dos fatores de regularização têm de ser um pouco diferentes porque, por exemplo, $\varepsilon_m = \|\boldsymbol{\eta}_m\|_2$, pode levar a um valor muito grande, sendo que $\|\boldsymbol{\eta}_m\|_2^2 = N_1 N_2 \sigma_m^2$, onde σ_m^2 é a variância do ruído gaussiano. Além do mais, pode-se estar sob uma situação onde o ruído é bem menor que o erro de movimento, causando uma grande diferença entre ε_m e os ε_k . Portanto, adota-se a seguinte situação:

1. Escolhe-se $\varepsilon_m=1$ e ε_k escolhido conforme (6.24), reescrito em (7.12).

$$\varepsilon_k^2 \geq \frac{\|\hat{\mathbf{e}}_{k,m}^I\|_2^2}{N_I}, \quad \text{se } \varepsilon_k^2 \geq \frac{\sigma_k^2}{\sigma_m^2}, \quad (7.12)$$

$$\varepsilon_k^2 = \frac{\sigma_k^2}{\sigma_m^2}, \quad \text{se } \varepsilon_k^2 < \frac{\sigma_k^2}{\sigma_m^2}$$

Esta escolha deve ponderar as imagens adicionais quanto aos erros de estimação de movimento e diferenças na variância do ruído.

2. Uma vez determinados os valores de ε_k , tem-se que:

$$\begin{aligned}
E^2 &= \|\mathbf{C}\mathbf{f}_m\|_2^2 = \mathbf{M}_1\mathbf{M}_2\sigma_C^2 = \mathbf{N}_1\mathbf{N}_2\mathbf{R}^2\sigma_C^2 \\
\lambda &= \frac{\kappa}{E^2} \\
\kappa &= \|\boldsymbol{\eta}\|_2^2 = \mathbf{L}\mathbf{N}_1\mathbf{N}_2\sigma_m^2, \quad \text{ou} \\
\kappa &= \|\boldsymbol{\eta}'\|_2^2 = \sigma_m^2 \sum_{k=1}^L \mathbf{N}_{lk},
\end{aligned} \tag{7.13}$$

onde \mathbf{N}_{lk} é o número de *pixels inliers* da imagem \mathbf{f}_k , \mathbf{R} é o fator de ampliação e σ_C^2 é a variância da imagem original filtrada pelo filtro passa-alta \mathbf{C} .

Em algumas situações, a variância do ruído pode ser estimada, porém a variância da imagem filtrada, não. Então, escolhe-se λ pensando na relação suavização do ruído/recuperação de detalhe. Essas escolhas permitem a realização prática do processo com bons resultados.

7.3. Compensação na Projecção do Método Iterativo com *Inliers*

Nesta seção, apresenta-se um problema muito comum na utilização de seqüência de imagens no processo iterativo, relacionado à dificuldade na convergência do método iterativo na medida em que mais quadros com movimento são utilizados.

Com a concatenação das imagens fornecidas pela seqüência, alguns autovalores nas matrizes $(\mathbf{D}_m^I)^T \mathbf{D}_m^I$ recebem uma contribuição maior que outros. Isso é esperado já que as regiões cuja informação foi retirada, devido aos *outliers* ou vetores mal-estimados, não contribuem para o aumento destes autovalores, aumentando a diferença entre os maiores e os menores autovalores de $(\mathbf{D}_m^I)^T \mathbf{D}_m^I$. Por isso, observa-se um efeito que faz com que a elipsóide da matriz concatenada, a *Bounding Ellipsoid*, se pareça menos com um elipsóide de revolução, ou esferóide, e torne-se mais irregular. É conhecido das técnicas de otimização que esse fenômeno prejudica a convergência [7]. Pode-se facilmente imaginar que um esferóide tem o negativo do gradiente apontado para o centro em todas as pontos, ao passo que um elipsóide irregular (achatado ou oblongo) tem o negativo do gradiente apontado para o centro apenas nas linhas que percorrem seus eixos.

Tem-se, então, que as regiões beneficiadas pela energia das outras imagens irão convergir primeiro em poucas iterações, sendo que as demais regiões terão dificuldades para convergir. A Figura 7.1 exemplifica o problema no caso bidimensional.

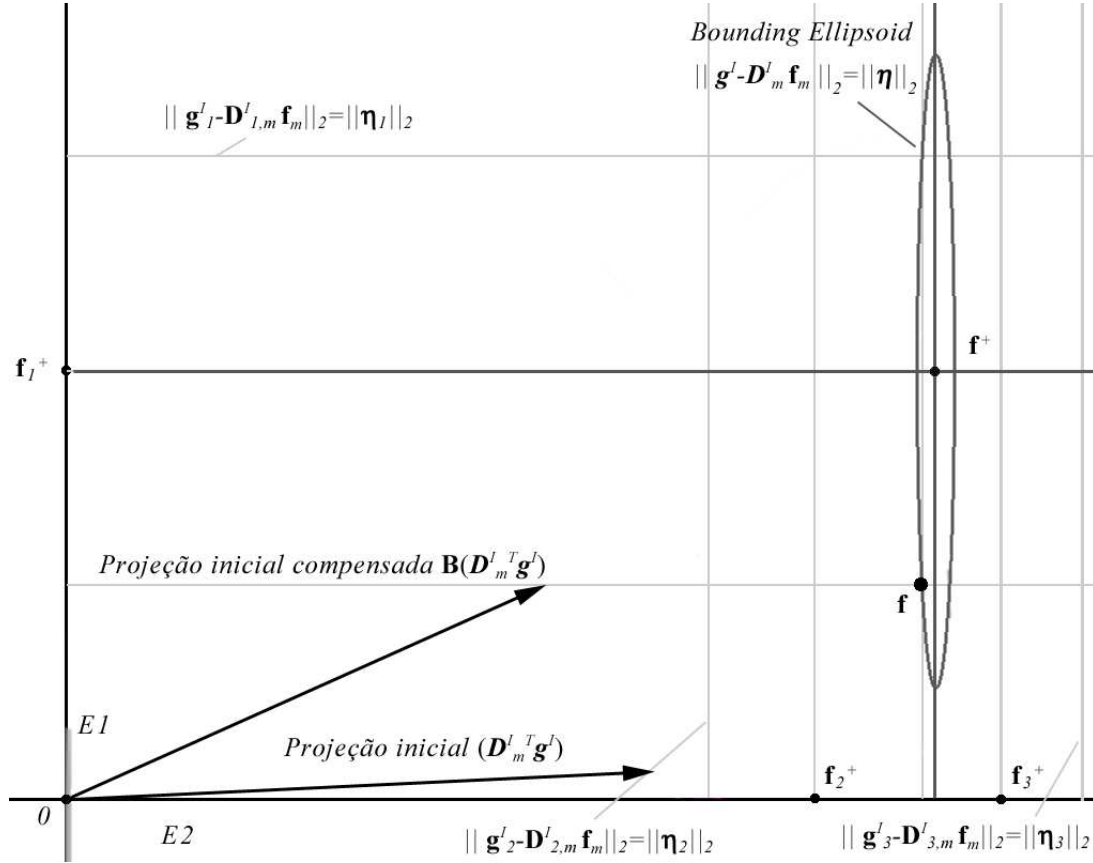


Figura 7.1: Problema no uso de seqüência de imagens no processo iterativo.

Nota-se que na Figura 7.1 a elipsóide torna-se muito maior num eixo do que no outro. Se as iterações iniciassem em zero, a projeção inicial iria aproximar a demonstrada na figura. Observa-se que, no Eixo $E1$, a otimização terá dificuldades de convergir. Então, uma compensação pode atenuar os efeitos da disparidade de energia entre as regiões, o que modifica a direção da projeção. A compensação faz com que a projeção reduza a energia nas componentes que foram exageradamente beneficiadas e aumente as que foram menos beneficiadas com a energia das várias imagens, dado que se conhece a energia média de contribuição em cada componente. A nova projeção apontará, de forma mais precisa, para o centro da elipsóide, aumentando a velocidade de convergência do método iterativo. O trabalho [26] propõe uma solução semelhante.

A compensação é apresentada como a matriz \mathbf{B} que acompanha o fator de relaxamento β , e está demonstrada abaixo:

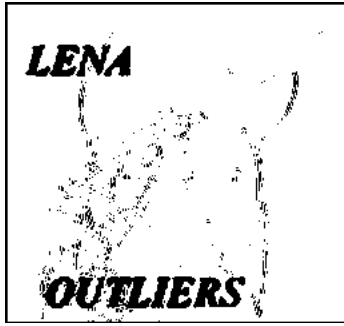
$$\mathbf{f}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta \mathbf{B} \left((\mathbf{D}_m^I)^T \mathbf{g}^I - ((\mathbf{D}_m^I)^T \mathbf{D}_m^I + \lambda \mathbf{C}^T \mathbf{C}) \mathbf{f}_m^{(n)} \right) \quad (7.14)$$

A matriz \mathbf{B} é uma matriz diagonal, de tamanho $M_1 M_2 \times M_1 M_2$, cujos elementos da diagonal principal são os pesos para compensação da projeção. Se os elementos da diagonal principal de \mathbf{B} forem representados através de um vetor \mathbf{b} , pode-se criar \mathbf{B} calculando \mathbf{b} . O vetor \mathbf{b} pode ser encontrado através de:

$$\mathbf{b} = \left(\left((\mathbf{D}_m^I)^T \mathbf{D}_m^I \right) \mathbf{1} \right)^{-1}, \quad (7.15)$$

onde $\mathbf{1}$ é um vetor $M_1 M_2 \times 1$ composto inteiramente por uns. Este vetor passa por $(\mathbf{D}_m^I)^T \mathbf{D}_m^I$ e os componentes do vetor resultante são invertidos individualmente; os componentes nulos não são invertidos.

Essa matriz de compensação equilibra a projeção acelerando a convergência. A Figura 7.2 demonstra em uma imagem o efeito da compensação na projeção inicial do método iterativo, onde a projeção inicial é $(\mathbf{D}_m^I)^T \mathbf{g}^I$.



Mapa de *outliers* estimados (em preto)



Projeção inicial,
 $\beta(\mathbf{D}_m^I)^T \mathbf{g}^I$



Projeção inicial compensada,
 $\beta \mathbf{B}(\mathbf{D}_m^I)^T \mathbf{g}^I$

Figura 7.2: Efeito na imagem da compensação na projeção do método iterativo.

Na Figura 7.2, a imagem que representa a projeção inicial foi formada por uma sequência de dois quadros. O mapa de *outliers*, em preto, demonstra a região onde *pixels* considerados *outliers* foram retirados de uma das imagens. Os efeitos na projeção inicial são as regiões mais escuras, com menos energia. Com a compensação da projeção, a convergência das diferentes regiões espaciais da imagem é mais uniforme.

Conclusão do Capítulo 7

Neste capítulo, foram apresentadas as modificações nas equações em razão da presença de *inliers* e *outliers* no movimento entre os quadros da sequência de imagens. Na primeira parte, foram demonstradas as mudanças no modelo de degradação, que passa a relacionar apenas parte das imagens da sequência de baixa resolução com a imagem desejada de alta resolução.

Na segunda parte deste capítulo, a mudança no equacionamento dos mínimos quadrados regularizados foi apresentada. As novas formulações do modelo de degradação foram incorporadas à técnica de restauração da resolução, o que permitiu que apenas os *inliers* fossem usados no algoritmo iterativo. Uma nova escolha das ponderações e dos fatores de regularização é sugerida.

Na última parte do capítulo, os problemas de convergência do algoritmo iterativo de sequência de imagens, devido à eliminação de informação dos *outliers* nos quadros adjacentes, são demonstrados. O uso de uma matriz de compensação das componentes da projeção é demonstrado como uma solução eficaz para auxiliar a convergência e aumentar a taxa de convergência.

Capítulo 8: Regularização Adaptativa para Sequência de Imagens

Neste capítulo, apresenta-se a regularização adaptativa para minimização das distorções causadas pela subamostragem. As distorções causadas pela subamostragem nem sempre são devidamente mencionadas e caracterizadas. Em muitos casos, isso ocorre devido à recuperação parcial dessa distorção através da utilização de sequência de imagens ou devido à superioridade da degradação causada pelo ruído. Mas em muitos sistemas de aquisição essa distorção é mais significativa, o que torna sua regularização necessária.

Nas seções seguintes, explicam-se a distorção causada pela subamostragem e a necessidade de regularização. Analisa-se o comportamento desse efeito em sequência de imagens com *inliers* e *outliers* e, ao final do capítulo, a proposta de regularização adaptativa é apresentada.

8.1. Regularização das Distorções Causadas pela Subamostragem

Analizando a degradação de única imagem, em (8.1) pode-se observar que o sistema linear é subdeterminado.

$$\mathbf{g}_m = \mathbf{D}_m \mathbf{f}_m + \boldsymbol{\eta}_m = \mathbf{S}_m \mathbf{H}_m \mathbf{f}_m + \boldsymbol{\eta}_m, \quad (8.1)$$

onde \mathbf{f}_m tem dimensões $M_1 M_2 \times 1$ e \mathbf{g}_m tem dimensões $N_1 N_2 \times 1$. Lembra-se que $N_1 N_2 < M_1 M_2$.

Na Equação (8.1) a subamostragem é a principal responsável pela subdeterminação do sistema porque a matriz \mathbf{S}_m , de tamanho $N_1 N_2 \times M_1 M_2$, é a causa da redução das dimensões do sinal. Segundo [8,23,25], a solução de Mínimos Quadrados pode falhar em condições de subdeterminação, o que gera distorções na imagem resultante. O trabalho [25] menciona que uma informação adicional deve ser utilizada na restauração de Mínimos Quadrados.

O uso de sequência de imagens traz benefícios, pois, à medida que as imagens adicionais são utilizadas, conforme (8.2), o sistema ganha mais equações, possivelmente trazendo mais informação sobre a imagem \mathbf{f}_m . Isso pode mudar as características do sistema de subdeterminado para determinado ou para sobredeterminado, tornando a distorção causada pela subamostragem menos expressiva.

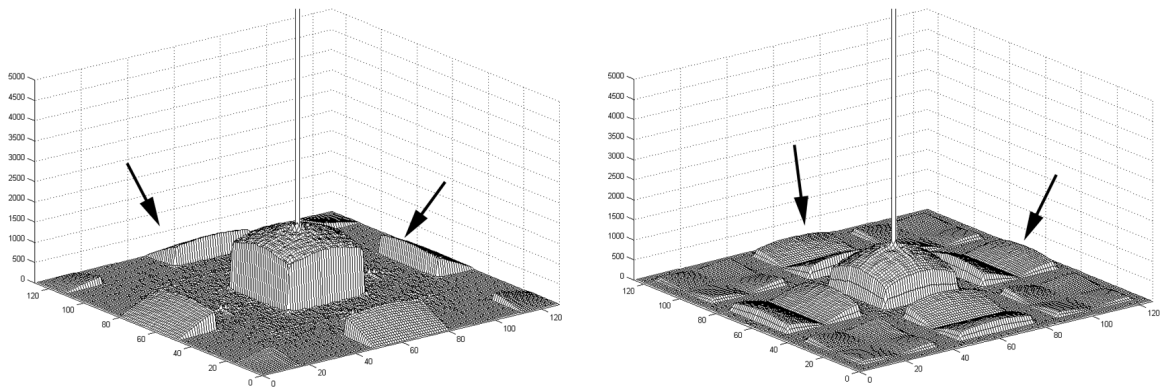
$$\begin{bmatrix} \mathbf{g}_1 \\ \varepsilon_1 \\ \vdots \\ \mathbf{g}_L \\ \varepsilon_L \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{1,m} \\ \varepsilon_1 \\ \vdots \\ \mathbf{D}_{L,m} \\ \varepsilon_L \end{bmatrix} \mathbf{f}_m + \begin{bmatrix} \boldsymbol{\eta}_1 \\ \varepsilon_1 \\ \vdots \\ \boldsymbol{\eta}_L \\ \varepsilon_1 \end{bmatrix}, \text{ ou } \mathbf{g} = \mathbf{D}_m \mathbf{f}_m + \boldsymbol{\eta}, \quad (8.2)$$

onde \mathbf{g} e $\boldsymbol{\eta}$ têm dimensões $LN_1 N_2 \times 1$ e \mathbf{D}_m tem dimensões $LN_1 N_2 \times M_1 M_2$.

Na maioria das aplicações, não se tem controle sobre a informação das imagens adicionais e não se pode garantir que a informação proporcionada por essas imagens remova as distorções da subamostragem. Matematicamente, pode-se dizer que as imagens adicionais podem não mudar a condição de subdeterminação do sistema linear. Por causa disso, a restauração de Mínimos Quadrados com seqüência de imagens ainda pode apresentar as distorções causadas pela subamostragem.

8.1.1. Distorções causadas por componentes de alta freqüência

O principal efeito da subdeterminação é o aparecimento de distorções na imagem estimada, causadas por componentes de alta freqüência gerados durante o processo iterativo. Isso ocorre devido à existência das etapas de subamostragem e sobreamostragem na matriz $\mathbf{D}_m^T \mathbf{D}_m$ em conjunto com a deficiência do filtro \mathbf{H}_m^T , onde $\mathbf{D}_m^T \mathbf{D}_m = \mathbf{H}_m^T \mathbf{S}_m^T \mathbf{S}_m \mathbf{H}_m$. Observe-se na Figura 8.1 o aparecimento dos componentes de alta freqüência. Nesta figura é mostrado o efeito na primeira iteração de $\mathbf{D}_m^T \mathbf{g}_m$. As setas na figura apontam os componentes gerados em virtude da subamostragem e que não foram devidamente atenuados pelo filtro \mathbf{H}_m^T .



(a) recuperação de resolução de duas vezes.

(b) recuperação de resolução de três vezes.

Figura 8.1: Aparecimento de componentes de alta freqüência devido à subamostragem.

Como mencionado em [8,25], a falha devido à subdeterminação é observada como o aparecimento desses componentes de alta freqüência, responsáveis pelas distorções na imagem estimada. A Figura 8.1 deixa claro que a subdeterminação é causada principalmente pela subamostragem.

O problema causado pela subamostragem pode ser corrigido com o uso de restrição, que é usada como a informação adicional mencionada por [8,25] para a solução de sistemas subdeterminados. A restrição deve minimizar os componentes de alta freqüência indesejados, reduzindo as distorções na imagem resultante.

8.1.2. Regularização dos efeitos da subamostragem

O aparecimento dos componentes de alta freqüência causa efeitos indesejados na imagem estimada; portanto, tem-se de regularizar o sistema, a fim de atenuar os componentes de alta freqüência. Dessa forma, o operador de regularização deve minimizar a energia na localização exata dos componentes de alta freqüência. A forma de aplicação da regularização é a mesma apresentada no Capítulo 4, conforme abaixo:

$$\hat{\mathbf{f}}_m^{RLS} = (\mathbf{D}_m^T \mathbf{D}_m + \alpha \mathbf{G}^T \mathbf{G})^{-1} \mathbf{D}_m^T \mathbf{g}_m, \quad (8.3)$$

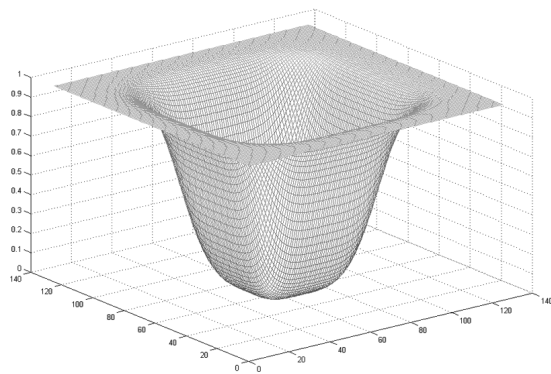
onde \mathbf{G} é uma matriz que representa o operador de regularização para os efeitos da subamostragem, que é diferente do antigo operador \mathbf{C} . O fator de regularização α descreve a proporção entre a energia dos componentes que surgem devido à subamostragem e a energia da imagem na mesma localização. A diferença entre na escolhas dos operadores \mathbf{C} e \mathbf{G} , quando se procura maximizar a recuperação do sinal e minimizar as distorções, são apresentados na Tabela 8.1.

Tabela 8.1: Tipos de operadores.

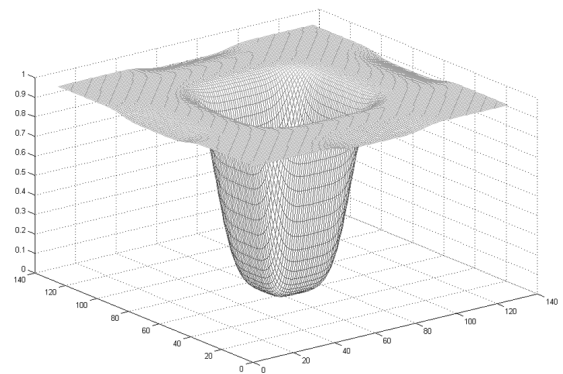
	Operador C	Operador G
Forma desejada do filtro:	$\frac{\text{Espectro de potência do ruído}}{\text{Espectro de potência do sinal}}$	$\frac{\text{Espectro de potência do efeito da subamostragem}}{\text{Espectro de potência do sinal}}$
Aproximado por:	Laplaciano, com freqüência de corte 1/2.	Filtro passa-alta mais seletivo (ex: spline cúbica), com freqüência de corte 1/R.

Na Tabela 8.1, R é o fator de ampliação, ou $R^2 = M_1 M_2 / N_1 N_2$. A forma desejado do filtro indica que se busca minimizar o termo no numerador, preservando o termo no denominador.

A Figura 8.2 demonstra o espectro do filtro escolhido como operador \mathbf{G} para os exemplos da Figura 8.1.



(a) exemplo de filtro para $R=2$.



(b) exemplo de filtro para $R=3$.

Figura 8.2: Exemplo do espectro de um filtro $\mathbf{G}^T \mathbf{G}$ escolhido como restrição.

Na Figura 8.3.a apresentam-se as distorções da subamostragem em imagem estimada e, na Figura 8.3.b, a imagem estimada usando a restrição \mathbf{G} , para $R=2$.



(a) imagem estimada com distorções.

(b) imagem estimada com distorções minimizadas.

Figura 8.3: Exemplo de imagens estimadas com distorções da subamostragem e regularização

No começo do capítulo foi mencionado que o uso de seqüência de imagens pode naturalmente corrigir o problema, o que será demonstrado na seção seguinte.

8.2. Seqüência de Imagens e Recuperação dos Efeitos da Subamostragem

Uma das informações mais importantes que as imagens adicionais podem trazer para o sistema é a informação sobre as amostras dizimadas no processo de subamostragem. Para que isso aconteça, o movimento entre imagens precisa coincidir com posições fracionárias, ou seja, o movimento deve ser de *subpixel*. Cada imagem adicional utilizada pode ser capaz de contribuir com uma nova informação, o que depende, sobretudo, do movimento. Então, imagens com movimentos de *subpixel* podem trazer a informação necessária para que o sistema deixe de ser subdeterminado. Se o sistema deixar de ser subdeterminado, os componentes de alta freqüência não aparecerão e não haverá necessidade de regularização. A Figura 8.4 demonstra casos em que os componentes de alta freqüência não aparecem em virtude da contribuição das imagens adicionais. Pode-se, então, dizer que a nova informação inverte ou cancela as distorções da subamostragem.

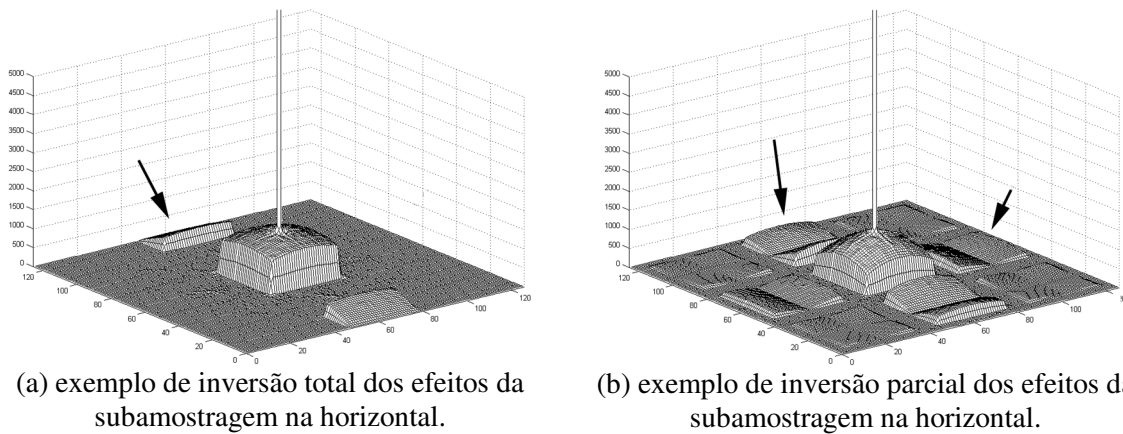


Figura 8.4: Inversão dos efeitos da subamostragem através do uso de seqüência de imagens.

O uso da expressão “inversão dos efeitos da subamostragem” através do uso de seqüência de imagens é válido porque os danos causados por esse processo, como um possível *aliasing*, são cancelados e os detalhes de alta freqüência da imagem são recuperados. Comparando as Figuras 8.1.a e a Figura 8.4.a, pode-se observar que os componentes de alta freqüência, no sentido horizontal, não aparecem, pois foram completamente cancelados. Já, na Figura 8.1.b e na Figura 8.4.b, o cancelamento foi apenas parcial e os componentes ainda são visíveis, mas com menos energia.

A regularização não é capaz de recuperar os detalhes na imagem estimada como o uso de seqüência de imagens, mas é capaz de atenuar as distorções causadas pela subamostragem, produzindo resultados mais significativos. Na seção seguinte, o problema será analisado de forma local.

8.3. Uso do Modelo com *Inliers* e *Outliers*

Conforme a teoria apresentada anteriormente, o uso de imagens adicionais pode inverter os efeitos da subamostragem. Porém, obter a quantidade suficiente de imagens com o movimento de *subpixel* necessário para inverter a subamostragem em toda a imagem é muito difícil em situações práticas. Isso ainda é agravado se houver *outliers* nas imagens da seqüência. Portanto, não se pode assumir que a seqüência terá toda a informação necessária, de modo que é esperado que apenas parte da imagem seja beneficiada com a informação de *subpixel*.

Então, o sistema terá regiões espaciais onde: os efeitos da subamostragem foram invertidos, onde foram parcialmente invertidos e onde se mantiveram. A Figura 8.5 demonstra um exemplo de inversão dos efeitos da subamostragem pelo uso de seqüência de imagens em recuperação de resolução de duas vezes, mas considerando a contribuição de *subpixel* de forma local.



Figura 8.5: Exemplo de inversão dos efeitos da subamostragem de forma local.

No exemplo da Figura 8.5, a região (a) foi beneficiada com contribuição diagonal, as regiões (b) e (c) receberam contribuições na horizontal e na vertical, respectivamente, e a região (d) não recebeu contribuição nenhuma.

Em virtude de a contribuição das imagens adicionais ser, na maioria das vezes, dispersa na imagem, a regularização dos efeitos de subamostragem tem de agir de forma local. É evidente que as regiões de *outliers* não trazem contribuição, mas nas regiões de *inliers* a contribuição depende do movimento de *subpixel*. Portanto, a contribuição proporcionada por cada imagem adicional precisa ser determinada e as posições de *outliers* e *inliers*, também. Assim, é possível prever onde os efeitos da subamostragem serão invertidos e onde será necessário usar a regularização.

Na seção seguinte, será apresentada a regularização adaptativa, demonstrando-se a análise da contribuição de cada imagem e a proposta deste trabalho para a solução do problema.

8.4. Regularização Adaptativa para os Efeitos da Subamostragem

Nas seções anteriores, foi introduzida a necessidade de regularização adaptativa dos efeitos da subamostragem em sequência de imagens. Nesta seção, a regularização adaptativa é detalhada. Esta regularização atua de forma local, atenuando as distorções causadas pela subamostragem.

Assume-se que cada imagem adicional pode contribuir conforme o vetor de deslocamento. Seguindo essa hipótese, a proposta de regularização deve considerar a energia de contribuição, a direção de contribuição e a posição espacial do vetor. Assim, pode-se determinar a regularização para cada região da imagem estimada. A regularização dos efeitos da subamostragem, demonstrada em (8.3), será utilizada nas equações tradicionais, mostradas em (7.10) e (7.11), na forma adaptativa. A nova proposta é apresentada em (8.4) e sua equação iterativa está apresentada em (8.5).

$$\hat{\mathbf{f}}_m^{ARLS} = \left((\mathbf{D}_m^I)^T \mathbf{D}_m^I + \lambda \mathbf{C}^T \mathbf{C} + \alpha \sum_{i=1}^P \mathbf{W}_i \mathbf{G}_i^T \mathbf{G}_i \right)^{-1} (\mathbf{D}_m^I)^T \mathbf{g}^I \quad (8.4)$$

$$\hat{\mathbf{f}}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta \left((\mathbf{D}_m^I)^T \mathbf{g}^I - \left((\mathbf{D}_m^I)^T \mathbf{D}_m^I + \lambda \mathbf{C}^T \mathbf{C} + \alpha \sum_{i=1}^P \mathbf{W}_i \mathbf{G}_i^T \mathbf{G}_i \right) \mathbf{f}_m^{(n)} \right) \quad (8.5)$$

Na Equação (8.4) e (8.5) α é o fator de regularização dos efeitos da subamostragem. A regularização adaptativa proposta é composta por um conjunto de P filtros \mathbf{G}_i . Cada um dos filtros \mathbf{G}_i é escolhido para proporcionar a atenuação de uma das componentes de alta frequência que surgem durante a subamostragem. No caso mais simples, de recuperação de resolução de duas vezes, $P=3$, onde se busca atenuar os três componentes de alta frequência: horizontal, vertical e diagonal. A matriz \mathbf{W}_i é uma matriz diagonal que representa os pesos do respectivo filtro \mathbf{G}_i em cada *pixel*. Uma fase inicial de análise de contribuição das imagens da seqüência pode determinar os valores das matrizes \mathbf{W}_i . Como o ruído ainda está presente, a regularização $\mathbf{C}^T \mathbf{C}$ é mantida.

As seções seguintes explicam como determinar os componentes da regularização adaptativa.

8.4.1. Filtros de regularização \mathbf{G}_i

Os filtros \mathbf{G}_i devem ser escolhidos para minimizar as componentes de alta frequência, exemplificadas nas Figuras 8.1 e 8.4, que surgem por causa da subamostragem. O aparecimento desses componentes no espectro pode ser apresentado como a adição de componentes moduladas do espectro da imagem. Então, P deve ser igual à quantidade de componentes moduladas. Por isso, a situação torna-se mais complexa à medida que o fator de ampliação aumenta, em razão do aparecimento de grande quantidade de componentes moduladas em diferentes frequências.

Considerando apenas o caso mais simples, como de recuperação de resolução de duas vezes, é possível o uso de um filtro em cada direção: horizontal, vertical e diagonal. Preferivelmente, a amplitude dos filtros deve ser normalizada, deixando mais simples a determinação de α .

A Figura 8.6 ilustra a o conjunto de filtros \mathbf{G}_i , considerando $P=3$ no caso em que $R=2$.

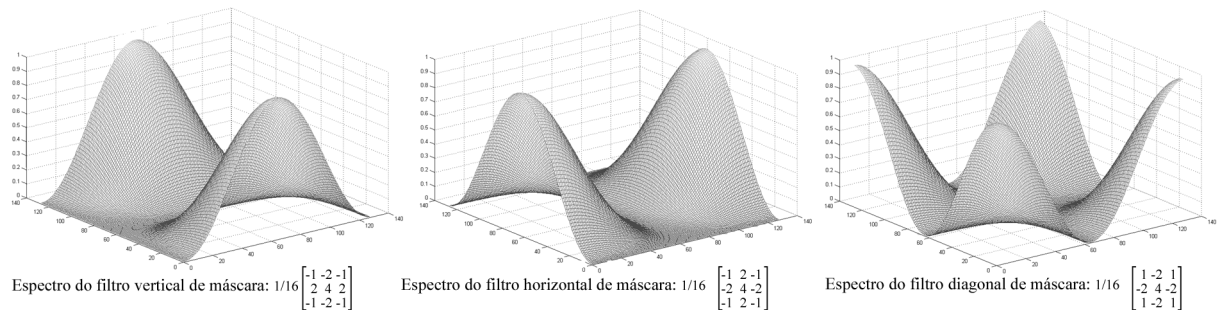


Figura 8.6: Espectro dos filtros da regularização adaptativa.

Nesse exemplo, os filtros são versões transformadas passa-alta do filtro passa-baixa Bilinear. O filtro Bilinear é uma aproximação discreta dos filtros *B-spline* de primeira ordem. *Splines* de ordem maior [19], como *Spline* Cúbica, além de outros tipos de filtros mais seletivos, podem

produzir resultados melhores; bons resultados são observados com *Splines* Cúbicas. Pode-se notar que as frequências de transformação dos filtros são as mesmas das componentes moduladas que surgiram na subamostragem.

8.4.2. Análise da contribuição das imagens e determinação de W_i e α

Para determinar os pesos W_i , é necessário saber a direção da contribuição de *subpixel* das imagens da seqüência. Como o movimento entre as imagens de alta resolução é conhecido, através da matriz de movimento $A_{k,m}$, conhecem-se os vetores de movimento e pode-se encontrar a fração de *subpixel*. Assim, se forem separadas as componentes horizontais e verticais dos vetores de deslocamento e divididas pelo fator de ampliação (R) escolhido, tem-se a informação fracionária, ou seja, de *subpixel*. Através da direção de *subpixel* de cada componente, é possível determinar o tipo de contribuição. A Figura 8.7 exemplifica a análise para a recuperação de resolução de duas vezes.

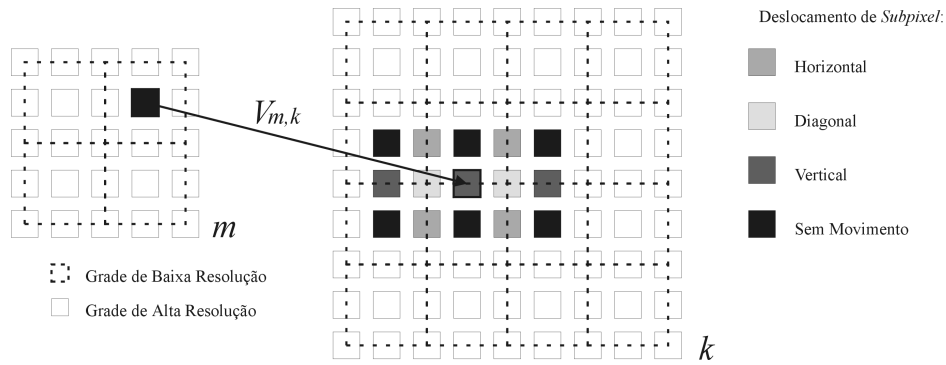


Figura 8.7: Exemplo da análise dos vetores para determinação da contribuição de *subpixel*.

Assume-se que cada vetor com deslocamento de *subpixel* diferente traz uma contribuição diferente. Para $R=2$, considerando $Y(z_1, z_2)$ o espectro de banda base (usando Transformada Z bidimensional), tem-se que cada imagem adicional traz a contribuição descrita em (8.6a-d), conforme sua direção de *subpixel*.

$$Y_C(z_1, z_2) = [Y(z_1, z_2) + Y(-z_1, z_2) + Y(z_1, -z_2) + Y(-z_1, -z_2)] / 4 \quad (8.6a)$$

$$Y_H(z_1, z_2) = [z_1 Y(z_1, z_2) - z_1 Y(-z_1, z_2) + z_1 Y(z_1, -z_2) - z_1 Y(-z_1, -z_2)] / 4 \quad (8.6b)$$

$$Y_V(z_1, z_2) = [z_2 Y(z_1, z_2) + z_2 Y(-z_1, z_2) - z_2 Y(z_1, -z_2) - z_2 Y(-z_1, -z_2)] / 4 \quad (8.6c)$$

$$Y_D(z_1, z_2) = [z_1 z_2 Y(z_1, z_2) - z_1 z_2 Y(-z_1, z_2) - z_1 z_2 Y(z_1, -z_2) + z_1 z_2 Y(-z_1, -z_2)] / 4 \quad (8.6d)$$

onde, z_1 representa o eixo horizontal e z_2 representa o eixo vertical. $Y_C(z_1, z_2)$ é a representação da transformada da imagem sem movimento de *subpixel*. $Y_H(z_1, z_2)$, $Y_V(z_1, z_2)$ e $Y_D(z_1, z_2)$ correspondem às representações da imagem com movimento de *subpixel* na horizontal, vertical e diagonal, respectivamente. Embora as equações representem deslocamentos globais, o problema pode ser avaliado localmente. A dedução completa dessas equações está no apêndice A1.

Assim, no caso de recuperação de resolução de duas vezes, em que são necessários três filtros, os pesos de cada um, em determinado *pixel*, serão:

$$\begin{aligned}
 W_C[m_1, m_2] &= E_C[m_1, m_2] + E_V[m_1, m_2] + E_H[m_1, m_2] + E_D[m_1, m_2] \\
 w_H[m_1, m_2] &= \left| \frac{E_C[m_1, m_2] + E_V[m_1, m_2] - E_H[m_1, m_2] - E_D[m_1, m_2]}{W_C[m_1, m_2]} \right|^2 \\
 w_V[m_1, m_2] &= \left| \frac{E_C[m_1, m_2] - E_V[m_1, m_2] + E_H[m_1, m_2] - E_D[m_1, m_2]}{W_C[m_1, m_2]} \right|^2 \\
 w_D[m_1, m_2] &= \left| \frac{E_C[m_1, m_2] - E_V[m_1, m_2] - E_H[m_1, m_2] + E_D[m_1, m_2]}{W_C[m_1, m_2]} \right|^2,
 \end{aligned} \tag{8.7}$$

onde $w_i[m_1, m_2]$ é o peso do filtro i ($i=H, V$ ou D) na posição $[m_1, m_2]$. $E_i[m_1, m_2]$ é o somatório das contribuições das imagens adicionais, cujo vetor de deslocamento na posição $[m_1, m_2]$ tem direção de *subpixel* i . Assume-se que cada imagem adicional tem contribuição igual a $\varepsilon_m/\varepsilon_k$, onde ε_k é a ponderação dada às imagens adicionais, a imagem base tem contribuição igual a $\varepsilon_m/\varepsilon_m=1$. $W_C[m_1, m_2]$ é a soma de todas as contribuições. Essas equações são desenvolvidas a partir das Equações (8.6a-d). Tanto as Equações (8.6a-d) quanto (8.7) devem ser desenvolvidas para cada fator de ampliação (R), considerando todas as componentes moduladas que surgem durante a subamostragem em questão. Os valores de $w_i[m_1, m_2]$ compõem a diagonal principal da matriz \mathbf{W}_i .

O valor do coeficiente de regularização (α) deve estar entre 0 e 1. Um método adequado para estimação ainda precisa ser desenvolvido. Nesse trabalho, é estimado de acordo com (8.8).

$$\alpha = \frac{\|\text{efeito indesejado atenuado (com } \mathbf{H}_m)\|}{\|\text{efeito indesejado sem atenuação (sem } \mathbf{H}_m)\|} \tag{8.8}$$

Essa forma leva $\alpha=1$ caso não exista um filtro \mathbf{H}_m , ou seja, os efeitos da subamostragem são os mais graves possíveis (porque \mathbf{H}_m não ajuda na atenuação as componentes de alta frequência que surgem por causa da subamostragem) e $\alpha=0$, caso o filtro \mathbf{H}_m seja um filtro ideal. A Equação (8.8) pode ser aproximada pela Equação (8.9), onde um espectro estimado de \mathbf{f}_m pode ser usado.

$$\alpha = \frac{\left\| \mathbf{D}_m^T \mathbf{D}_m \hat{\mathbf{f}}_m - \frac{1}{R^2} \mathbf{H}_m^T \mathbf{H}_m \hat{\mathbf{f}}_m \right\|}{\left\| \mathbf{S}_m^T \mathbf{S}_m \hat{\mathbf{f}}_m - \frac{1}{R^2} \hat{\mathbf{f}}_m \right\|}, \tag{8.9}$$

onde o fator $1/R^2$ deve equilibrar as amplitudes para que a banda base seja inteiramente extraída. Assim, apenas as altas frequências que surgiram com a subamostragem devem permanecer.

Nessas condições, a regularização deverá estar proporcionando apenas a suavização necessária para atenuar os efeitos indesejados da subamostragem. Os detalhes de alta resolução recuperados pelo uso de sequência de imagens são preservados.

8.5. Descrição do Processo Completo

Nesta seção, o processo completo de restauração da resolução de seqüência de imagens usando regularização adaptativa é ilustrado. A Figura 8.6 demonstra os processos envolvidos antes da realização da restauração iterativa.

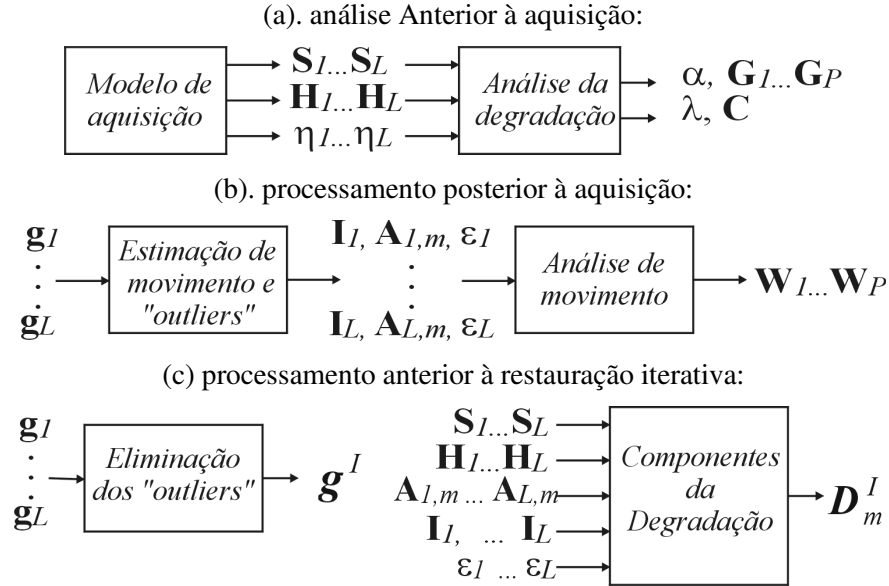


Figura 8.8: Etapas de processamento e análise anteriores ao processo iterativo.

Essas etapas são necessárias para a obtenção dos dados e dos parâmetros que são utilizados no processo iterativo. O diagrama de blocos da restauração iterativa está ilustrado na Figura 8.9.

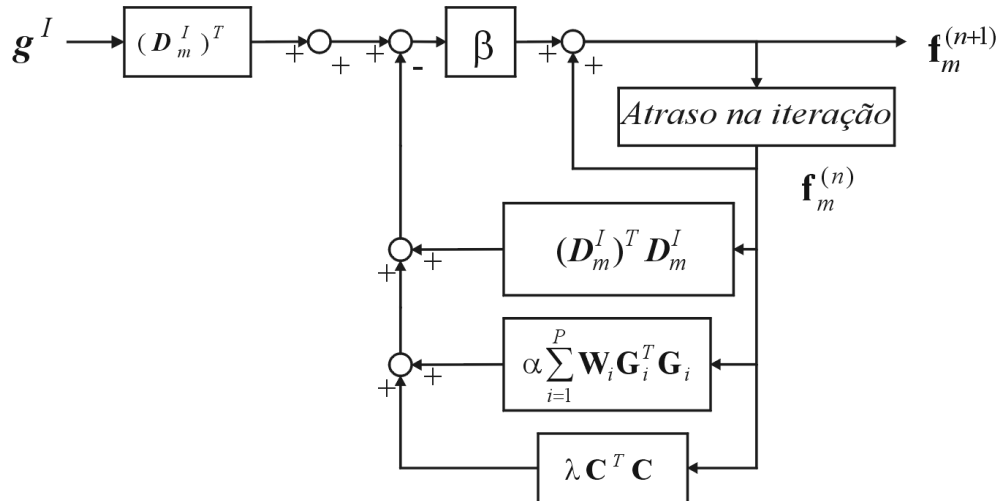


Figura 8.9: Diagrama de blocos do processo iterativo

O diagrama de blocos do processo iterativo mostra que o processo pode ser analisado como um sistema linear recursivo de primeira ordem, com múltiplas entradas e múltiplas saídas. No sistema, a seqüência de imagens é a informação inicial de entrada e, a cada iteração, parte da degradação da aquisição é reduzida, assim como as distorções da subamostragem e o ruído.

8.6. Acréscimo de Custo Computacional

O custo computacional acrescentado ao processo iterativo pode ser observado na Tabela 8.2. Os valores são aproximados, segundo informações de [28] e dos algoritmos implementados.

Tabela 8.2: Tabela com acréscimo de custo computacional.

Operação Principal	Iteração com Regularização Convencional	Iteração com Regularização Adaptativa
Processamento prévio:		
Estimação de movimento:	variável	variável
Estimação de outliers:	$2LM_1M_2$ FLOPS	$2LM_1M_2$ FLOPS
Determinação dos pesos:	zero	$R^2LM_1M_2$ FLOPS
Processo Iterativo:		
$\mathbf{b} = (\mathbf{D}_m^I)^T \mathbf{g}^I$	$n^2LM_1M_2$ FLOPS	$n^2LM_1M_2$ FLOPS
$\mathbf{c}_1 = ((\mathbf{D}_m^I)^T \mathbf{D}_m^I) \mathbf{f}_m^{(n)}$	$2n^2LM_1M_2$ FLOPS	$2n^2LM_1M_2$ FLOPS
$\mathbf{c}_2 = \lambda(\mathbf{C}^T \mathbf{C}) \mathbf{f}_m^{(n)}$	$2IM_1M_2$ FLOPS	$2IM_1M_2$ FLOPS
$\mathbf{c}_3 = \alpha(\mathbf{G}^T \mathbf{G}) \mathbf{f}_m^{(n)}$, ou $\alpha \sum_{i=1}^P (\mathbf{W}_i \mathbf{G}_i^T \mathbf{G}_i) \mathbf{f}_m^{(n)}$	$2IM_1M_2$ FLOPS	$4IPM_1M_2$ FLOPS
$\mathbf{f}_m^{(n+1)} = \mathbf{f}_m^{(n)} + \beta(\mathbf{b} - (\mathbf{c}_1 + \mathbf{c}_2 + \mathbf{c}_3))$	$6IM_1M_2$ FLOPS	$6IM_1M_2$ FLOPS
Custo adicionado: $R^2LM_1M_2 + (2-4P)IM_1M_2$		
M_1M_2 - Tamanho da imagem de alta resolução R - Fator de ampliação L - Número de imagens da sequência I - Número de iterações. P - Número de filtros base n - Dimensão da função de $h_k[m_1, m_2]$.		

O acréscimo computacional considera todas as operações como operações com ponto flutuante (FLOPS). A Figura 8.10 apresenta a relação entre custo computacional para solução e tamanho da imagem, onde são considerados $R=2$, $n=2$, $P=3$, $L=2$, com solução em 35 iterações, utilizando o algoritmo implementado.

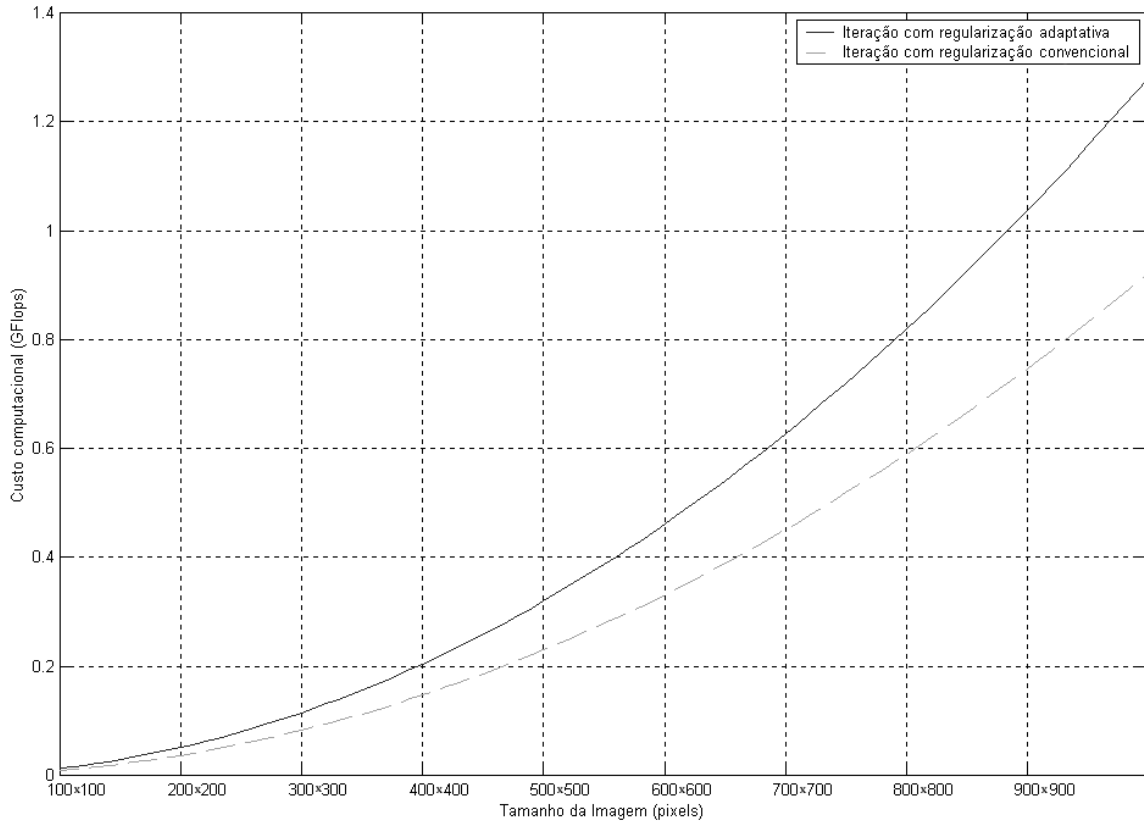


Figura 8.10: Custo computacional em relação ao tamanho das imagens.

Conclusões do Capítulo 8

Neste capítulo, demonstraram-se as distorções causadas pela subamostragem no método de Mínimos Quadrados. As distorções são causadas por componentes de alta frequência indesejados que surgem durante o processo iterativo. O problema pode ser solucionado com a atenuação das componentes indesejadas. As imagens com movimento de *subpixel* podem contribuir para o não-aparecimento das componentes indesejadas. Uma técnica adaptativa é proposta, a qual busca suavizar apenas as regiões que contêm as distorções e preservar as regiões onde houve contribuição.

A regularização adaptativa também considera a direção de contribuição, cuja análise nas direções horizontal, vertical e diagonal foi feita. Os pesos para os filtros adaptativos foram estudados para ampliação de duas vezes. A regularização adaptativa visa proporcionar uma suavização adequada para cada região, buscando, assim, produzir resultados melhores que os obtidos pelos métodos não adaptativos.

Capítulo 9: Resultados Experimentais Simulados

Neste capítulo, demonstram-se os resultados experimentais da teoria apresentada. Na simulação inicial, todas as etapas do processo de restauração da resolução são demonstradas, incluindo:

- geração de movimento artificial e de *outliers* artificiais;
- degradação sintética;
- estimativa de única imagem para estimação de movimento;
- estimação do movimento gerado artificialmente;
- estimação da possível região de *outliers*;
- determinação dos filtros \mathbf{C} e \mathbf{G}_i , $i=1,\dots,P$, para aplicação no caso simulado;
- determinação dos fatores de regularização: ε_k , para $k=1,\dots,L$, e também λ e α ;
- determinação dos pesos \mathbf{W}_i , $i=1,\dots,P$, para o conjunto de filtros \mathbf{G}_i ;
- processo iterativo e determinação de β ;
- comparação de resultados.

Os problemas de cada etapa são comentados e comparados com os da teoria. Na segunda simulação, apenas alguns pontos específicos serão comentados.

9.1. Simulação 1

Nesta simulação é utilizada a imagem ‘Lenna’, com 200×200 *pixels*, apresentada na Figura 9.1.a. Esta imagem será chamada de \mathbf{f}_m , e é parte de uma imagem maior de 260×260 . As amplitudes são 0 para a cor preta e 255 para a cor branca, com 256 variações em tons de cinza.

9.1.1. Geração de movimento artificial e *outliers* artificiais

Partindo da imagem maior, uma segunda imagem, com movimento translacional global de um *pixel* horizontal e um *pixel* vertical, é criada. Em uma região será colocada uma tarja branca para representar os *outliers*. A imagem é apresentada na Figura 9.1.b e será chamada de \mathbf{f}_k .



(a) original

(b) com movimento e *outliers* artificiais**Figura 9.1:** Imagens da seqüência de alta resolução.

As informações sobre o movimento entre as imagens e sobre os *outliers* são encontradas no Capítulo 3.

9.1.2. Degradação artificial

A seqüência irá sofrer uma redução de resolução de duas vezes. A degradação óptica será uma operação de média móvel (MA), de máscara retangular, entre dois *pixels* horizontais e dois verticais (2×2). A subamostragem será com grade retangular, onde $\Delta m_1 = \Delta m_2 = 2$. O conjunto de distorção óptica e subamostragem pode ser representado por:

$$g_k[n_1, n_2] = \sum_{m_1=2n_1, m_2=2n_2}^{2n_1+1, 2n_2+1} \frac{1}{4} f_k[m_1, m_2] \quad (9.1)$$

Nesta simulação não será adicionado ruído. A degradação será a mesma para a imagem f_m e para a imagem f_k . As imagens degradadas estão representadas pelas Figuras 9.2.a e 9.2.b.

(a) imagem g_m .(b) imagem g_k .**Figura 9.2:** Imagens da seqüência degradada artificialmente.

As informações sobre o modelo de degradação estão no Capítulo 2.

9.1.3. Estimativa de única imagem e estimação do movimento artificial

Para a estimação de única imagem será utilizada interpolação bicúbica. As duas imagens com estimativa de única imagem são usadas para estimar movimento. No modelo de movimento translacional global, apenas dois parâmetros precisam ser estimados (deslocamento no eixo m_1 e no eixo m_2). A Figura 9.3 demonstra a superfície de erro de estimação $\|e_{k,m}\|^2$ em função dos parâmetros.

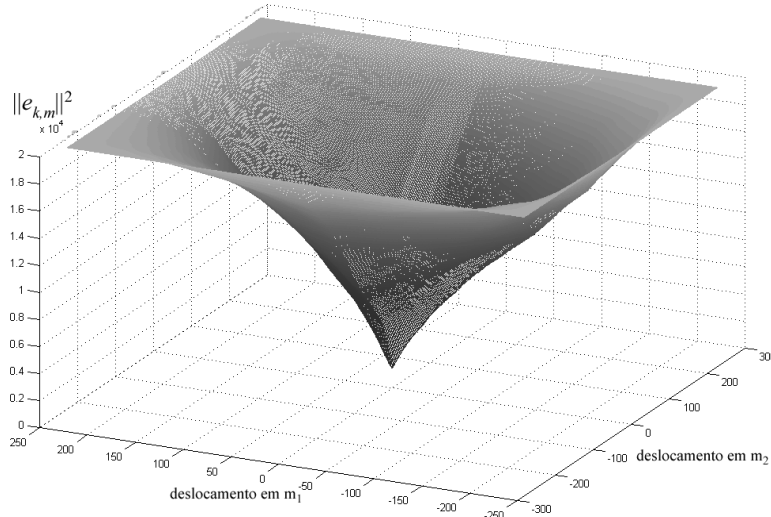
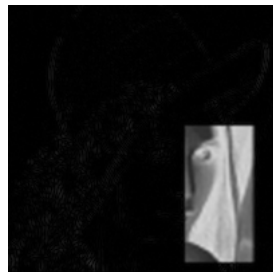


Figura 9.3: Superfície formada pela norma quadrática do erro de estimação.

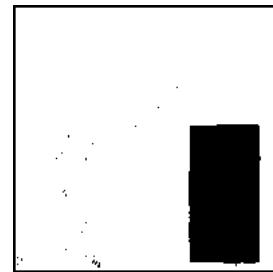
O resultado da estimação dos parâmetros de $\mathbf{A}_{m,k}(\boldsymbol{\theta})$ foi $\boldsymbol{\theta}=[1,1]$, que corresponde ao deslocamento de um *pixel* horizontal e um *pixel* vertical, igual ao movimento gerado artificialmente. As informações sobre a estimação de movimento estão no Capítulo 6.

9.1.4. Estimativa da região de outliers

A máscara de *outliers* é estimada a partir do erro entre a imagem \mathbf{f}_m e a imagem compensada $\mathbf{A}_{m,k}\mathbf{f}_k$. A Figura 9.4.a mostra o erro entre as imagens. Amplitudes absolutas acima de um limiar $T=\sigma_{DFD}$ são consideradas *outliers*. A Figura 9.4.b mostra a máscara resultante.



(a) erro entre as imagens compensadas, $\mathbf{e}_{k,m}$.



(b) máscara estimada de *outliers*, \mathbf{i}_k .

Figura 9.4: Estimação dos *outliers*.

Maiores informações sobre a estimação dos *outliers* são encontradas no Capítulo 6.

9.1.5. Determinação dos filtros C e G_i

Como não se está considerando ruído nesta simulação, não há necessidade da determinação de um filtro C. Para minimizar as distorções causadas pela subamostragem, é utilizado um conjunto de três filtros ($P=3$). Os filtros são baseados em *Spline* Cúbica e estão ilustrados na Figura 9.5.

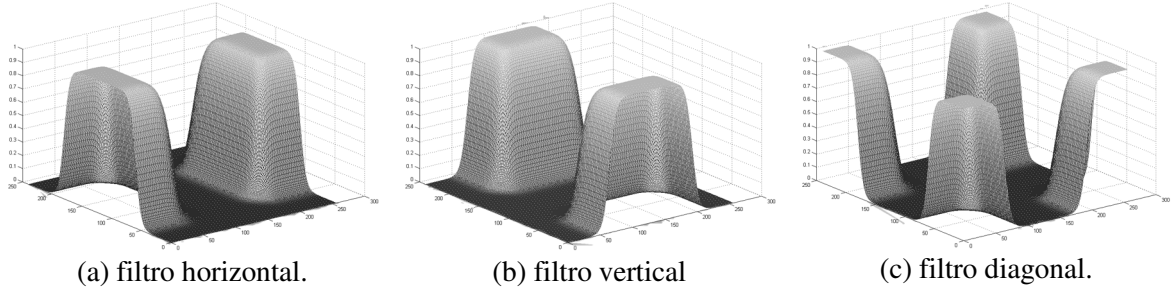


Figura 9.5: Espectro dos filtros G_i .

Informações sobre a determinação dos filtros são encontradas no Capítulo 8.

9.1.6. Determinação dos fatores de regularização e ponderações

A determinação das ponderações ϵ_k , para $k=1\dots L$, será feita conforme a teoria mencionada no Capítulo 7:

$$\begin{aligned}\epsilon_m^2 &= 1 \\ \epsilon_k^2 &= \frac{\|\hat{\mathbf{e}}_{k,m}^I\|^2}{N_I} \\ \lambda &= 0\end{aligned}\tag{9.2}$$

O fator de regularização α é escolhido segundo a Equação (8.9). Para este experimento, tem-se que: $\epsilon_m=1$, $\epsilon_k=8$, $\lambda=0$, e $\alpha=0,1$. Maiores informações sobre a determinação dos fatores de regularização são encontradas nos Capítulos 4, 6, 7 e 8.

9.1.7. Determinação dos pesos W_i para os filtros G_i

Os pesos dos filtros são computados segundo a Equação (8.7). A imagem \mathbf{f}_m proporciona uma contribuição uniforme em todos os *pixels* $E_C[m_1, m_2] = \epsilon_m / \epsilon_m = 1$. A imagem adicional proporciona uma contribuição maior na horizontal e na vertical, porque seu movimento de *subpixel* é diagonal, onde $E_D[m_1, m_2] = \epsilon_m / \epsilon_k = 1/8$. Os valores de ϵ_m e ϵ_k são as ponderações determinadas na seção anterior. A Figura 9.6 apresenta os pesos na forma de máscaras, onde branco é peso um, com tons intermediários até preto, que é peso zero.

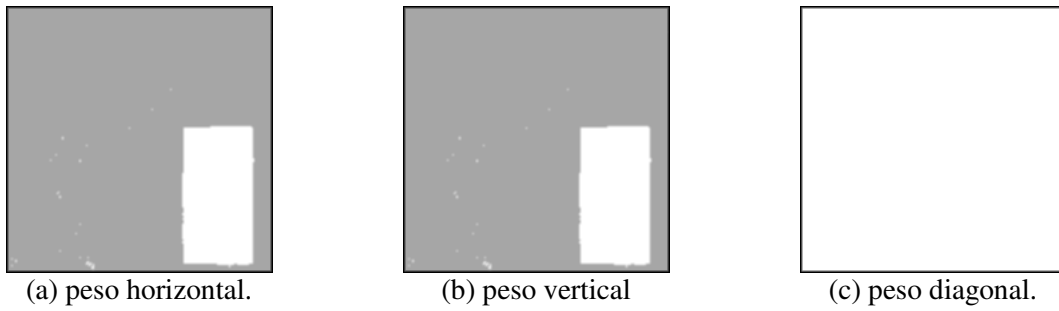


Figura 9.6: Pesos W_i dos filtros G_i .

Mas informações sobre a escolha dos pesos são vistos no Capítulo 8 e no Apêndice A1.

9.1.8. Processo iterativo e determinação de β

Para a interpolação iterativa é preciso escolher o método de busca e determinar um critério de parada. Se for escolhido o método de busca de Distanciamento Simultâneo, é preciso determinar β . Nesta simulação, será utilizada a busca de Descida mais Íngreme; portanto, não há necessidade de determinar β .

Para critério de parada, pode-se escolher um valor mínimo para a norma da projeção, ou um valor mínimo para qualquer componente da projeção; pode-se também definir um número máximo de iterações. Para esta simulação o critério de parada será a norma arredondada igual a zero, segundo (9.3), ou máximo de 50 iterações:

$$\text{round}\left(\left\|\Phi(\mathbf{f}_m^{(n)})\right\|_2\right)=0, \quad (9.3)$$

onde $\text{round}(x)$ é uma função de arredondamento simétrico, onde x entre 0,5 e 1 passa a ser 1 e x entre 0 e 0,499 passa a ser zero. Isso garante que todas as componentes tenham erro menor ou igual a 0,5 (lembra-se que as amplitudes finais da imagem são discretas, entre 0 e 255). A compensação de projeção proposta no Capítulo 7 também foi utilizada.

9.1.9. Comparação dos resultados

Para demonstrar o desempenho das técnicas, será feita uma comparação visual e uma comparação objetiva. Para comparação objetiva serão utilizados o PSNR (*Peak Signal-to-Noise Ratio*) e o DSNR (*Differential Signal-to-Noise Ratio*), segundo:

$$\text{PSNR} = 10 \log_{10} \left(\frac{255^2}{\frac{1}{M_1 M_2} \|\mathbf{f} - \mathbf{f}_e\|^2} \right) \quad \text{e} \quad \text{DSNR} = 10 \log_{10} \left(\frac{\|\mathbf{f} - \mathbf{f}_0\|^2}{\|\mathbf{f} - \mathbf{f}_e\|^2} \right), \quad (9.4)$$

onde \mathbf{f} é a imagem original; \mathbf{f}_e , a imagem estimada e \mathbf{f}_0 , a imagem interpolada, segundo a interpolação de ordem zero.

As técnicas utilizadas para comparação de resultados são:

1. Interpolação bilinear, segundo [1].
2. Interpolação bicúbica, segundo [19].
3. Interpolação regularizada com única imagem (usando apenas \mathbf{g}_m), segundo:

$$\hat{\mathbf{f}}_m = ((\mathbf{D}_m^T \mathbf{D}_m + \lambda \mathbf{C}^T \mathbf{C} + \alpha \mathbf{G}^T \mathbf{G})^{-1} \mathbf{D}_m^T \mathbf{g}_m \quad (9.5)$$

4. Interpolação não-regularizada com seqüência de imagens, segundo:

$$\hat{\mathbf{f}}_m = ((\mathbf{D}_m^I)^T \mathbf{D}_m^I)^{-1} (\mathbf{D}_m^I)^T \mathbf{g}^I \quad (9.6)$$

5. Interpolação com regularização não-adaptativa com seqüência de imagens, segundo:

$$\hat{\mathbf{f}}_m = ((\mathbf{D}_m^I)^T \mathbf{D}_m^I + \lambda \mathbf{C}^T \mathbf{C} + \alpha \mathbf{G}^T \mathbf{G})^{-1} (\mathbf{D}_m^I)^T \mathbf{g}^I \quad (9.7)$$

6. Interpolação com regularização adaptativa com seqüência de imagens (a técnica proposta), segundo:

$$\hat{\mathbf{f}}_m = ((\mathbf{D}_m^I)^T \mathbf{D}_m^I + \lambda \mathbf{C}^T \mathbf{C} + \alpha \sum_{i=1}^P \mathbf{W}_i \mathbf{G}_i^T \mathbf{G}_i)^{-1} (\mathbf{D}_m^I)^T \mathbf{g}^I \quad (9.8)$$

Embora as interpolações 3-6 estejam escritas na forma de inversão de matrizes, são realizadas na forma iterativa.

Como não foi considerado o ruído, a restrição \mathbf{C} não é utilizada, sendo o fator de regularização λ igual a zero. A comparação objetiva está na Tabela 9.1.

Tabela 9.1: Comparação dos resultados.

Método de Ampliação:	PSNR	DSNR
Interpolação bilinear	26.988 dB	0.380 dB
Interpolação bicúbica	27.517 dB	0.909 dB
Interpolação iterativa regularizada com única imagem , Equação (9.5).	28.302 dB	1.695dB
Interpolação iterativa não-regularizada com seqüência de imagens ($\alpha=0$), Equação (9.6).	29.635 dB	3.027 dB
Interpolação iterativa com regularização não-adaptativa , Equação (9.7).	29.990dB	3.383dB
Interpolação iterativa com regularização adaptativa , Equação (9.8).	30.472 dB	3.864 dB



(a) interpolação bilinear.



(b) interpolação bicúbica.



(c) interpolação iterativa regularizada de única imagem.



(d) interpolação iterativa não-regularizada com seqüência de imagens.



(e) interpolação iterativa com regularização não-adaptativa e com seqüência de imagens.



(f) interpolação iterativa com regularização adaptativa e com seqüência de imagens.

Figura 9.7: Resultados visuais das técnicas.

Para comparação visual, os resultados são mostrados na Figura 9.7. Pode-se notar que os métodos de restauração da resolução (interpolação iterativa, Figuras 9.7.c, 9.7.d, 9.7.e, e 9.7.f) produzem resultados superiores aos das interpolações convencionais (Figuras 9.7.a e 9.7.b). Isso comprova que a restauração da resolução, mesmo de única imagem, recupera a degradação. Mesmo sem o ruído, a regularização é necessária para reduzir as distorções da subamostragem. Pode-se observar na Figura 9.7.d que na região em que não houve benefício do quadro adicional, devido aos *outliers*, as distorções da subamostragem são bem visíveis (região onde foi colocada a tarja). Na Figura 9.7.e, a regularização adaptativa minimiza apenas as distorções da subamostragem preservando os detalhes recuperados na região de contribuição da imagem adicional.

A simulação seguinte apresenta um caso com movimento simulado, mas com *outliers* reais. A degradação artificial, agora, conterá ruído.

9.2. Simulação 2

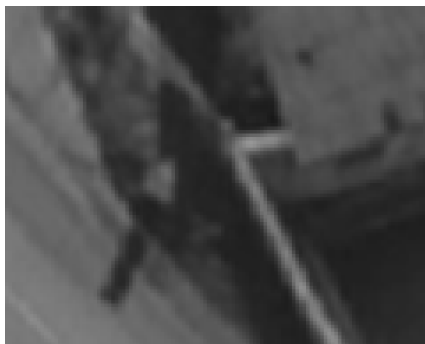
Nesta simulação é apresentada uma situação mais real, sendo consideradas quatro imagens, apresentadas na sequência da Figura 9.8. As imagens possuem movimento global, porém a imagem na Figura 9.8.b possui apenas translação; a imagem na Figura 9.8.c possui escalamento e a imagem na Figura 9.8.d possui rotação. O movimento foi gerado artificialmente, mas a região de *outliers* (o veículo e bordas) é natural.



(a) imagem 01, f_m .



(b) imagem 02, com translação.



(c) imagem 03, com escalamento.



(d) imagem 04, com rotação.

Figura 9.8: Imagens da sequência de alta resolução.

O movimento é pequeno para que a estimação seja mais rápida e segura (já que o propósito do trabalho não é avaliar as técnicas de estimação de movimento). Segundo a teoria do Capítulo 3, esses movimentos podem ser modelados com o movimento tipo *affine*, de seis parâmetros.

A degradação sofrida pelas imagens da sequência é a seguinte:

- degradação óptica através de um filtro espaço-invariante de média móvel retangular de dimensões 2×2 ;
- subamostragem de duas vezes, em grade retangular;
- adição de ruído gaussiano de 5 dB.

Essa degradação é uma aproximação discreta da degradação de um CCD em recuperação de resolução de duas vezes [2]. A sequência degradada está apresentada na Figura 9.9.

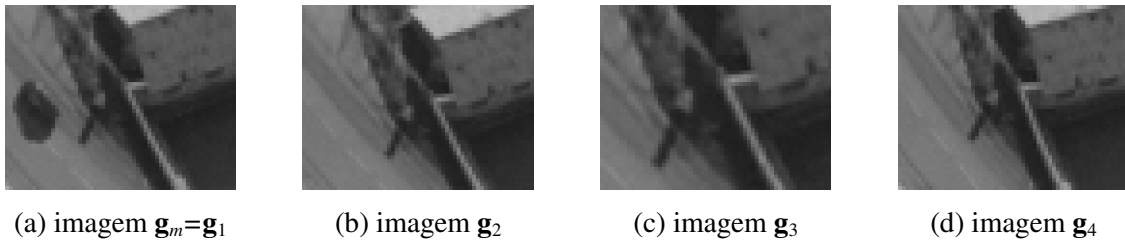


Figura 9.9: Sequência degradada.

Este experimento tenta recuperar a imagem f_m , usando, primeiramente, apenas a imagem g_m , em seguida, g_m e g_2 , e, por último, as quatro imagens da sequência. As explicações das etapas do processo de interpolação iterativa são omitidas, visto que são exatamente as mesmas demonstradas na simulação anterior. A Tabela 9.2 apresenta a comparação dos resultados numéricos obtidos e a Figura 9.10 mostra a comparação visual.

Na Tabela 9.2, o fator de regularização λ diferente de zero implica que a regularização para o ruído está sendo usada. O fator de regularização α diferente de zeros implica que a regularização dos efeitos da subamostragem está sendo utilizada. A tabela também menciona a equação que é utilizada na obtenção do respectivo resultado.

A Tabela 9.2 traz várias informações novas. Os resultados demonstram que apenas a regularização para o ruído, em muitos casos, é insuficiente. A Tabela 9.2 também demonstra que muitas imagens com pouca informação nova e com ruído não contribuem muito para a melhoria dos resultados.

Como ambas as restrições, C e G , que são funções passa-alta, é esperado que, quanto maior for o valor do respectivo fator de regularização, maior será a suavização da imagem estimada. Então, os fatores de regularização controlam a variação entre uma imagem com características suaves (poucos detalhes e poucas distorções) e uma imagem com características bruscas (mais detalhes e mais distorções). Portanto, um fator de regularização pequeno permite a recuperação de

detalhes na estimativa, contudo, se houver muito ruído e pouca contribuição a estimativa conterá muitas distorções.

Tabela 9.2: Comparação o dos resultados.

Método de Ampliação:	PSNR	DSNR
Interpolação bilinear .	35.688 dB	1.993 dB
Interpolação bicúbica . (Figura 9.10.a)	37.133 dB	3.545 dB
Interpolação iterativa regularizada com única imagem ($\lambda=0,005$ e $\alpha=0,1$), Equação (9.5). (Figura 9.10.b)	40.089 dB	6.394 dB
Interpolação iterativa não regularizada , com dois quadros (g_m e g_2) ($\lambda=0$ e $\alpha=0$), Equação (9.6). (Figura 9.10.c)	37.884 dB	4.189 dB
Interpolação iterativa regularizada , com dois quadros (g_m e g_2) ($\lambda=0,01$ e $\alpha=0,0$), Equação (9.7).	41.075 dB	7.379 dB
Interpolação iterativa regularizada , com dois quadros (g_m e g_2) ($\lambda=0,00$ e $\alpha=0,1$), Equação (9.7).	41.992 dB	8.296 dB
Interpolação iterativa regularizada , com dois quadros (g_m e g_2) ($\lambda=0,01$ e $\alpha=0,1$), Equação (9.7).	41.953 dB	8.257 dB
Interpolação iterativa regularizada adaptativa , com dois quadros (g_m e g_2) ($\lambda=0,01$ e $\alpha=0,1$), Equação (9.8). (Figura 9.10.d)	42.232 dB	8.537 dB
Interpolação iterativa não regularizada , com todos os quadros ($\lambda=0$ e $\alpha=0$), Equação (9.6). (Figura 9.10.e)	37.878 dB	4.183 dB
Interpolação iterativa regularizada , usando todos quadros ($\lambda=0,02$ e $\alpha=0,0$), Equação (9.7).	41.800 dB	8.104 dB
Interpolação iterativa regularizada , usando todos quadros ($\lambda=0,0$ e $\alpha=0,1$), Equação (9.7).	42.595 dB	8.899 dB
Interpolação iterativa regularizada , usando todos quadros ($\lambda=0,02$ e $\alpha=0,1$), Equação (9.7).	42.579 dB	8.884 dB
Interpolação iterativa regularizada adaptativa , usando todos quadros ($\lambda=0,02$ e $\alpha=0,1$), Equação (9.8). (Figura 9.10.f)	42.670 dB	8.975 dB



(a) interpolação bicúbica.



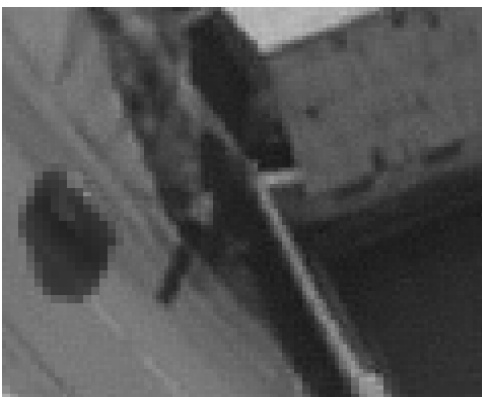
(b) interpolação iterativa regularizada de única imagem.



(c) interpolação iterativa não regularizada, com dois quadros.



(d) interpolação iterativa regularizada adaptativa, com dois quadros.



(e) interpolação iterativa não regularizada com quatro quadros.



(f) interpolação iterativa com regularização adaptativa e com quatro quadros.

Figura 9.10: Resultados visuais das técnicas.

Através da comparação visual das imagens na Figura 9.10, pode-se observar que as imagens produzidas pelas técnicas não regularizadas apresentam efeitos de bloco (distorção causada pela subamostragem) nas regiões sem contribuição. Portanto, a regularização adaptativa tenta proporcionar a regularização necessária para minimizar as distorções da subamostragem, evitando atenuar possíveis detalhes recuperados.

Conclusões do Capítulo 9

Nesse capítulo foram apresentados resultados experimentais simulados. Essas simulações permitem uma comparação objetiva (numérica), por meio do DSNR e PSNR, e subjetiva (visual) dos resultados. Os objetivos deste capítulo foram:

- demonstrar as etapas envolvidas no processo de restauração da resolução;
- mostrar que a restauração da resolução (interpolação iterativa) pode produzir resultados melhores que a interpolação convencional;
- demonstrar a vantagem do uso de sequência de imagens na restauração da resolução;
- comprovar que a regularização adaptativa pode obter resultados ainda melhores, pois preserva os detalhes recuperados, minimizando as distorções da subamostragem de forma local.

Assim, os experimentos simulados sustentam a teoria apresentada e comprovam os benefícios da técnica proposta por este trabalho.

Capítulo 10: Aplicações em Casos Reais

No capítulo anterior, realizaram-se experimentos simulados para comprovar a teoria apresentada por esta dissertação; neste capítulo, as técnicas são utilizadas em casos reais, demonstrando seu potencial em aplicações práticas. As técnicas são aplicadas na restauração da imagem de alta resolução de uma sequência adquirida por uma câmera digital comercial.

No primeiro caso, é demonstrada a aquisição de imagens de um conjunto de prédios, com o objetivo apenas de produzir uma imagem com melhor resolução. No segundo caso, as técnicas são utilizadas para melhorar a informação de texto contida em uma placa, a qual está ilegível por causa da falta de resolução. O uso de restauração de resolução deve tornar o texto legível, possibilitando recuperar a informação.

10.1. Zoom de Uma Cena Estática

Neste caso, será utilizada uma sequência de seis imagens e a resolução será aumentada em duas vezes. Como não se conhece a degradação real, será assumida a degradação discreta de um CCD para a recuperação de resolução de duas vezes (a mesma usada nas simulações do capítulo anterior).

A Figura 10.1 ilustra os resultados da recuperação de resolução de duas vezes, mostrando a imagem de baixa resolução escolhida como \mathbf{g}_m (com interpolação de Ordem Zero) e as imagens ampliadas através dos métodos iterativos de sequência de imagens, com regularização não adaptativa e com regularização adaptativa. A região de *outliers*, neste caso, são as bordas; o movimento é assumido como translacional global.

Através da Figura 10.1 observam-se alguns pontos interessantes. A regularização não adaptativa, da Figura 10.1.b, com $\lambda=0,05$ e $\alpha=0,1$, suaviza exageradamente a imagem para minimizar as distorções da subamostragem. Na Figura 10.1.c, com $\lambda=0,05$ e $\alpha=0,0$, o efeito na área com contribuição é melhor, porém observam-se distorções nas bordas, decorrentes da falta de contribuição das imagens adicionais e da falta de regularização para minimização das distorções. Já, na Figura 10.1.d, as regiões de borda são mais suavizadas e a região mantém a informação recuperada, permitindo um aspecto visual melhor.



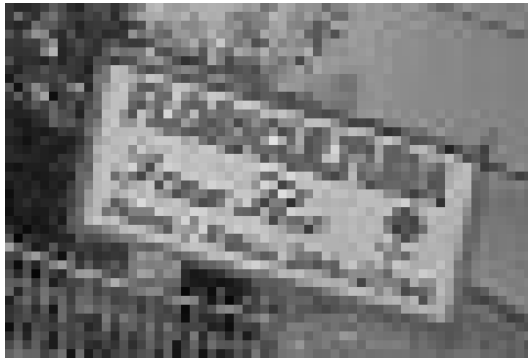
(a) imagem adquirida.

(b) interpolação regularizada. ($\lambda=0,05$ e $\alpha=0,1$)(c) interpolação regularizada. ($\lambda=0,05$ e $\alpha=0,0$)(d) regularização adaptativa. ($\lambda=0,05$ e $\alpha=0,1$)**Figura 10.1:** Recuperação da resolução de duas vezes.

10.2. Melhoria de Definição de Textos

Neste caso, a informação contida em uma placa de identificação de um estabelecimento está ilegível por causa da falta de resolução da câmera. A técnica será utilizada na tentativa de tornar a informação legível a um observador humano. Os mesmos modelos de aquisição e de movimento dos casos anteriores serão assumidos para uma recuperação de resolução de duas vezes. Quatro imagens são utilizadas para a restauração da resolução. Para comparação visual, uma outra aquisição é feita de uma distância mais próxima. As imagens estão na Figura 10.2.

A análise visual demonstra que qualquer uma das técnicas com sequência de imagens já torna o texto principal, em sua maior parte, legível, mas a interpolação iterativa com regularização adaptativa torna a imagem mais parecida com a imagem adquirida em uma distância mais próxima.



(a) imagem adquirida.



(b) interpolação bicúbica.

(c) interpolação regularizada. ($\lambda=0,01$ e $\alpha=0,0$)(d) interpolação regularizada. ($\lambda=0,01$ e $\alpha=0,5$)(e) interpolação adaptativa ($\lambda=0,01$ e $\alpha=0,5$)

(f) imagem adquirida em uma distância mais próxima.

Figura 10.2: Ampliação de duas vezes.

Conclusões do Capítulo 10

Neste capítulo, foi possível demonstrar que a técnica é realizável em situações práticas. Devido a limitações das técnicas de estimação de movimento, apenas situações com movimentos globais, tipo translacional ou *affine*, foram demonstradas.

Conclusões e Discussões Futuras

Neste capítulo final, as conclusões dos capítulos anteriores são recordadas. A idéia geral sobre restauração da resolução utilizando seqüência de imagens é concluída. Em seguida, são expostas sugestões para a continuidade da evolução desta tecnologia.

Conclusões Finais

Nesta dissertação, foi demonstrado como uma imagem de alta resolução pode ser estimada a partir de uma ou mais de uma seqüência de imagens adquiridas. Cada capítulo abrange uma parte do problema, apresentando possibilidades e limitações do processo.

No Capítulo 2, o modelo de formação de imagens, que descreve a formação de imagens nos dispositivos de aquisição, foi apresentado. O modelo de formação de imagens pode ser visto como a degradação da imagem de alta resolução durante a captura. Concluiu-se que o modelo linear é adequado para representar a degradação em uma grande quantidade de aplicações.

No Capítulo 3, os modelos do movimento entre as imagens foram apresentados, permitindo que as imagens da seqüência possam ser usadas na restauração. Com a informação das imagens adicionais, uma imagem de melhor qualidade pode ser produzida. Porém, os modelos de movimento 2d são limitados e não representam movimentos reais 3D muito complexos. Concluiu-se que as seqüências de imagens que contenham *outliers* podem ser usadas se os mesmos forem detectados e retirados dos dados.

No Capítulo 4, a formulação da solução para restauração da resolução foi apresentada. Como o modelo da degradação é linear, a solução de sistemas lineares foi utilizada. O desenvolvimento da solução de sistemas lineares mal-condicionados foi apresentado, através do método de Mínimos Quadrados Regularizados (RLS), tendo-se concluído que o uso de conjuntos teóricos ponderados (BEM) é uma das escolhas mais adequadas para seqüência de imagens.

No Capítulo 5, o método iterativo para solução dos sistemas lineares foi apresentado. Como a solução dos sistemas envolve inversão de matrizes e as matrizes da degradação de seqüência de imagens são extremamente grandes, a solução através de iteração é uma realizável. As vantagens do método iterativo, bem como duas realizações possíveis, foram apresentadas.

No Capítulo 6, a estimação de movimento a partir das imagens degradadas e a determinação dos *outliers* foram discutidas. Concluiu-se que pequenos erros durante o processo de estimação de movimento são esperados em razão do uso de imagens de baixa resolução na estimativa de movimento.

No Capítulo 7, a teoria foi modificada para as aplicações práticas considerando *inliers* e *outliers*. As modificações no modelo de degradação foram mostradas e diretamente incorporadas

na proposta de solução. Um método prático para a escolha das ponderações e do fator de regularização é proposto e, inclusive, uma compensação de projeção para o processo iterativo.

No Capítulo 8, a principal contribuição desta dissertação foi exposta. Na maioria dos casos, a nova informação e a contribuição proporcionada pelas imagens adicionais não é uniforme. Essa nova informação é dependente da posição espacial e da direção dos vetores de movimento que relacionam imagens adicionais à imagem desejada. A nova informação contribui para a o não-aparecimento das distorções causadas pela subamostragem e também ajuda a recuperar detalhes na imagem estimada. Por outro lado, a regularização acaba atenuando os detalhes recuperados. Então, propõe-se uma regularização adaptativa, que busca preservar os detalhes recuperados nas regiões de contribuição e atenuar as distorções nas regiões onde não houve contribuição.

No Capítulo 9, as simulações que confirmam a teoria de base e a proposta de melhoramento da técnica através da regularização adaptativa foram apresentadas. Os resultados são comparados de forma subjetiva (visual) e objetiva (numérica).

No Capítulo 10, algumas aplicações nas quais a técnica pode trazer benefícios foram demonstradas. Porém, apenas resultados visuais subjetivos foram expostos.

Por fim, esta dissertação é concluída apresentando-se três pontos principais observados nas técnicas de restauração da resolução de imagens:

1. a restauração da resolução pode recuperar a degradação causada durante a aquisição, proporcionando melhores resultados que as interpolações convencionais;
2. o uso de sequência de imagens traz benefícios para a restauração da resolução. Se as imagens adicionais tiverem informação de *subpixel*, as distorções causadas pela subamostragem podem ser atenuadas e os detalhes podem ser recuperados;
3. a regularização adaptativa pode proporcionar a suavização necessária em cada região da imagem, devido à diferença de contribuição de *subpixel* na imagem. A regularização adaptativa obtém resultados melhores que uma regularização não adaptativa.

Esses comentários concluem a dissertação; em sequência, apresentam-se os pontos e direções a serem desenvolvidos em trabalhos futuros.

Discussões Futuras

Nesta seção são discutidas propostas de melhoramento da técnica observadas ao longo do período de pesquisa. Linhas de solução não adotadas são mencionadas, com as possíveis vantagens citadas, bem como algumas propostas de solução não continuadas.

Quanto ao movimento entre as imagens, ficaram claras as limitações de representação do movimento real 3D pelo movimento aparente 2D. É possível prever se as aplicações podem, ou não, ser beneficiadas pelas técnicas de sequência de imagens. Todavia, mesmo que haja um modelo

de movimento 2D que possa ser utilizado para representar o movimento 3D em uma sequência de imagens, estimá-lo não é simples. Algumas deduções sobre as dificuldades teóricas na estimação de movimento são apresentadas no Capítulo 6, mas uma revisão das técnicas de estimação de movimento fica fora do alcance deste trabalho. Técnicas robustas de estimação de movimento são um assunto à parte, ainda mais quando se busca movimento em nível de *subpixel* [13-14,22]. Por essa razão, simulações de aplicações com movimentos complexos (com muitos parâmetros), como seqüências de vídeo com pessoas, por exemplo, não foram apresentadas. A estimação de *inliers* e *outliers* pode ser mais bem desenvolvida. Nessa pesquisa, apenas métodos básicos que permitem a continuidade do trabalho foram utilizados.

Quanto ao modelo de degradação, foi observado que o modelo linear é uma aproximação aceitável e que, inclusive, pode representar outros tipos de sistemas de formação de imagem [1]. A solução de sistemas lineares por Mínimos Quadrados já está muito bem desenvolvida. Há muitos trabalhos já desenvolvidos no campo de restauração de imagens que podem ser usados em restauração de resolução, inclusive técnicas que aparentam ser muito melhores para essa aplicação. Exemplos desses são: Mínimos Quadrados Ponderados (WLS – *Weighted Least Squares*) [27], potencialmente melhor em aplicações com *outliers*; Mínimos Quadrados Totais (TLS – *Total Least Squares*) [20], possivelmente melhor em razão da existência de erros na matriz de degradação e na matriz de movimento. Quanto à regularização, também existem bastantes soluções já apresentadas [1], porém a idéia de considerar a adaptação de acordo com a contribuição de *subpixel* não foi previamente encontrada na literatura.

O método iterativo, para busca da solução do sistema, também está bem desenvolvido. Existem algoritmos iterativos com alta taxa de convergência [21] e métodos de busca mais sofisticados [3]. Este trabalho apresenta apenas os métodos básicos, para que todas as etapas envolvidas na restauração da resolução sejam observadas.

Como observação final, tem-se que uma das principais necessidades para melhoramento das técnicas atuais é de técnicas de estimação de movimento rápidas e robustas, que possam ser aplicadas em situações de movimento complexo e gerem vetores confiáveis. Se isso for conseguido, diversas aplicações práticas podem ser beneficiadas com a restauração da resolução de seqüência de imagens.

Apêndice A1: Subamostragem Regular

Neste anexo, apresenta-se a informação matemática sobre as contribuições de *subpixel* em seqüência de imagens. O conteúdo desta seção é apresentado partindo do caso unidimensional, e em seguida, o caso bidimensional. Este material segue os princípios da decomposição em fases (*Polyphase Decompositions*). Mais informação sobre este material é encontrado em [1,6,9].

A1.1. Caso Unidimensional, Subamostragem de Duas Vezes

Inicialmente, considera-se que, um sinal $y[m]$ amostrado sofrerá uma subamostragem de duas vezes. A subamostragem pode ser vista como duas etapas: dizimação (ou cancelamento) de amostras regularmente espaçadas e mudança na taxa de amostragem. A dizimação é apresentada em (A.1a), e a mudança na taxa de amostragem é apresentada em (A.1b).

$$y_{D0}[m] = \frac{1}{2} (y[m] + (-1)^m y[m]) \quad (\text{A.1a})$$

$$y_{S0}[n] = y_{D0}[m/2], \quad m \text{ par} \quad (\text{A.1b})$$

onde $y_{D0}[n]$ é a seqüência após a dizimação e $y_{S0}[n]$ com a taxa de amostragem reduzida.

Se a seqüência $y[m]$ fosse atrasada em 1 amostra e então fosse subamostrada, uma outra seqüência, a $y_{S1}[n]$, seria criada. Esta nova seqüência contém as amostra que foram dizimadas na geração de $y_{S0}[n]$. Observe em (A.2):

$$y_{D1}[m] = \frac{1}{2} (y[m+1] + (-1)^m y[m+1]) \quad (\text{A.2})$$

$$y_{S1}[n] = y_{D1}[m/2], \quad m \text{ par}$$

A Figura A.1 ilustra os sinais subamostrados em fases diferentes.

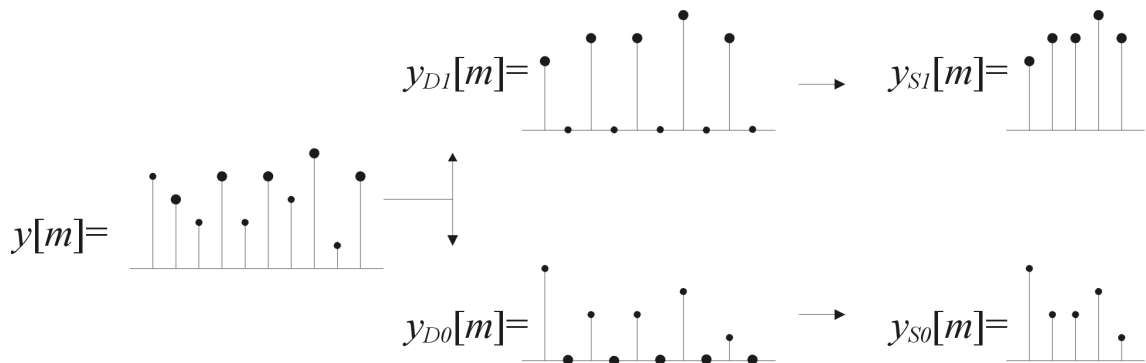


Figura A.1.1: Sinal ao sofrer subamostragem.

Pode-se representar esse processo no plano Z. Neste caso, vê-se que as Transformadas Z dos dois sinais estão relacionados, segundo (A.3).

$$\begin{aligned}
Y_{D0}[z] &= \frac{1}{2} (Y[z] + Y[-z]) \\
Y_{S0}[z] &= \frac{1}{2} (Y[z^{1/2}] + Y[-z^{1/2}]) \\
e \\
Y_{D1}[z] &= \frac{1}{2} (zY[z] - zY[-z]) \\
Y_{S1}[z] &= \frac{1}{2} (z^{1/2}Y[z^{1/2}] - z^{1/2}Y[-z^{1/2}]),
\end{aligned} \tag{A.3}$$

onde o termo z elevado a $1/2$ significa a redução em duas vezes na taxa de amostragem. A multiplicação de $Y[\cdot]$ por z significa uma defasagem de uma amostra, e a multiplicação por $z^{1/2}$ a defasagem de meia amostra.

Analisando a representação no plano Z , pode-se tirar duas conclusões:

1. Se o sinal não sofreu *aliasing* durante a subamostragem, então a componente modulada $Y[-z]$ não invadiu a região de banda base $Y[z]$. Assim, retornando a taxa de amostragem e aplicado uma filtragem passa-baixa a componente $Y[-z]$ é completamente eliminada, sem distorções em $Y[z]$.
2. Se o sinal sofreu *aliasing* na subamostragem, então a recuperação só pode ser feita com os dois sinais subamostrados $y_{S0}[n]$ e $y_{S1}[n]$, da forma apresentada na Equação (A.4).

Observe que no primeiro caso qualquer uma das duas seqüências subamostradas, $y_{S0}[n]$ ou $y_{S1}[n]$, recuperaria o sinal original $y[m]$ (desde que a fase de $y_{S1}[n]$ fosse compensada). Porém, no segundo caso, os dois sinais são necessários, conforme (A.4):

$$\begin{aligned}
Y[z] &= Y_{S0}[z^2] + z^{-1}Y_{S1}[z^2] \\
Y[z] &= \frac{1}{2} (Y[z] + Y[-z]) + z^{-1} \frac{1}{2} (zY[z] - zY[-z])
\end{aligned} \tag{A.4}$$

A Equação (A.4) demonstra que o sinal $y_{S1}[n]$, após retornada a taxa de amostragem e compensado na defasagem, pode recuperar o sinal original ao ser adicionado ao sinal $y_{S0}[n]$, também com a taxa de amostragem retornada, invertendo os efeitos de um possível *aliasing*.

Este princípio adotado para recuperar o sinal é o mesmo adotado na recuperação dos detalhes da imagem através da informação de *subpixel*.

A1.2. Caso Bidimensional, Subamostragem de Duas Vezes

No caso bidimensional, tem-se que um sinal bidimensional discreto $y[m_1, m_2]$, ao ser subamostrado por duas vezes, recebe a adição de componentes moduladas nas duas dimensões, segundo (A.5):

$$\begin{aligned}
y_{D00}[m_1, m_2] &= \frac{1}{4} (y[m_1, m_2] + (-1)^{m_1} y[m_1, m_2] + (-1)^{m_2} y[m_1, m_2] + (-1)^{m_1+m_2} y[m_1, m_2]) \\
y_{S00}[n_1, n_2] &= y_{D00}[m_1/2, m_2/2], \quad \text{para } m_1 \text{ e } m_2 \text{ pares}
\end{aligned} \tag{A.5}$$

De forma similar, outros três sinais subamostrados são formados. Na Figura A.2 a formação de $y_{S00}[m_1, m_2]$ é ilustrada.

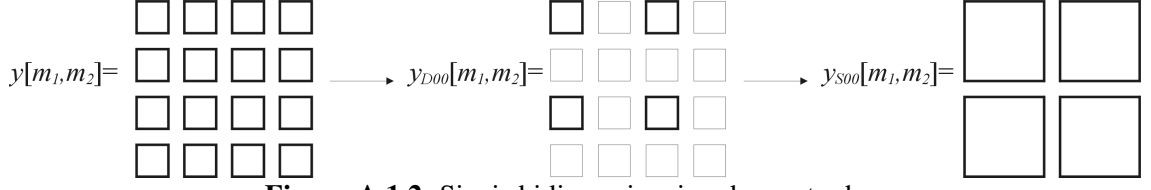


Figura A.1.2: Sinais bidimensionais subamostrados.

Pode-se representar a Transformada Z dos sinais, demonstrando sua relação com o sinal original, conforme (A.6):

$$\begin{aligned}
 Y_{S00}[z_1^{1/2}, z_2^{1/2}] &= \frac{1}{4} (Y[z_1^{1/2}, z_2^{1/2}] + Y[-z_1^{1/2}, z_2^{1/2}] + Y[z_1^{1/2}, -z_2^{1/2}] + Y[-z_1^{1/2}, -z_2^{1/2}]) \\
 Y_{S10}[z_1^{1/2}, z_2^{1/2}] &= \frac{1}{4} (z_1^{1/2} Y[z_1^{1/2}, z_2^{1/2}] - z_1^{1/2} Y[-z_1^{1/2}, z_2^{1/2}] + z_1^{1/2} Y[z_1^{1/2}, -z_2^{1/2}] - z_1^{1/2} Y[-z_1^{1/2}, -z_2^{1/2}]) \\
 Y_{S01}[z_1^{1/2}, z_2^{1/2}] &= \frac{1}{4} (z_2^{1/2} Y[z_1^{1/2}, z_2^{1/2}] + z_2^{1/2} Y[-z_1^{1/2}, z_2^{1/2}] - z_2^{1/2} Y[z_1^{1/2}, -z_2^{1/2}] - z_2^{1/2} Y[-z_1^{1/2}, -z_2^{1/2}]) \\
 Y_{S11}[z_1^{1/2}, z_2^{1/2}] &= \frac{1}{4} (z_1^{1/2} z_2^{1/2} Y[z_1^{1/2}, z_2^{1/2}] - z_1^{1/2} z_2^{1/2} Y[-z_1^{1/2}, z_2^{1/2}] - z_1^{1/2} z_2^{1/2} Y[z_1^{1/2}, -z_2^{1/2}] + z_1^{1/2} z_2^{1/2} Y[-z_1^{1/2}, -z_2^{1/2}])
 \end{aligned} \tag{A.6}$$

onde z_1 é referente a m_1 e z_2 é referente a m_2 . Pode-se dizer que se $Y_{S00}[z_1, z_2]$ é a transformada Z sinal subamostrado central, então $Y_{S10}[z_1, z_2]$ possui a informação de *subpixel* no eixo m_1 (por exemplo, horizontal), $Y_{S01}[z_1, z_2]$ possui a informação de *subpixel* no eixo m_2 (por exemplo, vertical) e $Y_{S11}[z_1, z_2]$ possui a informação de *subpixel* correlacionada nos eixos m_1 e m_2 (por exemplo diagonal).

Da mesma forma que no caso unidimensional, se não houve *aliasing* qualquer um dos sinais pode retornar o sinal original. Porém, se houve *aliasing* em qualquer um dos eixos serão necessários os sinais com a informação de *subpixel* para recuperar o *aliasing*. (Ver Equação (A.7)).

$$Y[z_1, z_2] = Y_{S00}[z_1^2, z_2^2] + z_1^{-1} Y_{S10}[z_1^2, z_2^2] + z_2^{-1} Y_{S01}[z_1^2, z_2^2] + z_1^{-1} z_2^{-1} Y_{S11}[z_1^2, z_2^2] \tag{A.7}$$

Como o *aliasing* acontece por causa das informações de alta frequência, então se está recuperando esta informação de alta frequência. Portanto, pode-se dizer que os detalhes da imagem estão sendo recuperados.

Embora tenha sido apresentado o caso retangular, de subamostragem de duas vezes, outros arranjos e resoluções podem ser analisados pelos mesmos princípios.

Referências Bibliográficas

Livros

- [1] BOVIK, A. *Handbook of Image and Video Processing*, Academic Press, 2000.
- [2] JÄHNE, B., HAUBEKER, H. e GEIBLER, P. *Handbook of Computer Vision and Applications: vol. 1, Sensors and Imaging*, Academic Press, 1999.
- [3] GONZALEZ, R. C. e WOODS, R. E. *Digital Image Processing*, Addison Wesley Longman, 1992.
- [4] NOBLE, B. e DANIEL, J. W. *Álgebra Linear Aplicada*, Prentice-Hall do Brasil, 1997.
- [5] KREYSZIG, E. *Advanced Engineering Mathematics*, 8th edition, John Wiley & Sons, 1999.
- [6] OPPENHEIM, A. V., WILLSKY A. S. e NAWAB, S. H. *Signals and Systems*, 2nd edition, Prentice-Hall, 1997.
- [7] HIMMELBLAU, D. M. *Applied Nonlinear Programming*, McGraw-Hill, 1972.
- [8] HANSEN, P. C. *Rank-Deficient and Ill-Posed Problems*, SIAM, 1997.
- [9] OPPENHEIM, A. V., SCHAFER, R. W. e BUCK, J. R. *Discrete-Time Signal Processing*, 2nd edition, Prentice-Hall, 1998.

Periódicos

- [10] TOM, B. C. e KATSAGGELOS, A. K. "Resolution enhancement of monochrome and color video using motion compensation", *IEEE Transactions on Image Processing*, **vol. 10 no. 2**, p. 278-287, Fevereiro de 2001.
- [11] SCHULTZ, R. R. e STEVENSON, R. L. "Extraction of high-resolution frames from video sequences", *IEEE Transactions on Image Processing*, **vol. 5 no. 6**, p. 996-1010, Junho de 1996.
- [12] SHIN, J. H., PAIK, J. K., PRICE, J. R. e ABIDI, M. A. "Adaptive regularized image interpolation using data fusion and steerable constraints", *Proc. SPIE Visual Comm. Image Proc.*, **vol. 4310**, p. 798-808, Janeiro de 2001.
- [13] STILLER, C. e KONRAD, J. "Estimating motion in image sequences", *IEEE Signal Processing Magazine*, **16**, p. 70-91, Julho de 1999.
- [14] SCHULTZ, R. R., MENG, L. e STEVENSON, R. L. "Subpixel motion estimation for super-resolution image sequence enhancement", *Journal of visual communication and image representation*, **vol. 9, no. 1**, p. 38-50, Março de 1998.

- [15] KATSAGGELOS, A. K. "Iterative image restoration algorithms", *Optical engineering, special issue on Visual Communications and Image Processing*, vol. **28**, no. **7**, p. **735-748**, Julho de 1989.
- [16] KATSAGGELOS, A. K., SCHAFER R. W. e MERSEREAU, R. M. "A regularized iterative image restoration algorithm," *IEEE Trans. on Signal Processing*, vol. **39**, no. **4**, p **914-929**, Abril de 1991.
- [17] GALATSANOS, N. P. KATSAGGELOS, A. K., CHIN, R. T. e HILLERY, A. D. "Least squares restoration of multi-channel images", *IEEE Trans. on Signal Processing*, vol. **39**, no. **10**, p. **2222-2236**, Outubro de 1991.
- [18] ELAD, M. e FEUER, A. "Restoration of single superresolution image from several blurred, noisy and undersampled measured images", *IEEE Trans. on Image Processing*, vol. **6**, no. **12**, p. **1646-1654**, Dezembro de 1997.
- [19] UNSER, M. "Splines: a perfect fit for signal and image processing", *IEEE Signal Processing Magazine*, **16**, p. **22-38**, Novembro de 1999.
- [20] MESAROVIC, V. Z., GALATSANOS, N. P. e KATSAGGELOS, A. K. "Regularized constrained total least-squares image restoration", *IEEE Trans. Image Processing*, vol. **4**, no. **8**, p. **1096-1108**, Agosto de 1995
- [21] KATSAGGELOS, A. K., e EFSTRATIADIS, S. N. "A Class of iterative signal restoration algorithms", *IEEE Trans. Acoustic, Speech and Signal Processing*, vol. **38**, no. **5**, p. **778-786**, Maio de 1990.

Teses e Revisões

- [22] HASLER, D. *Perspectives on Panoramic Photography*, Ph.D. Thesis, EPFL, nb. **1419**, Julho de 2001.
- [23] SNEDDON, W. G. *Smoothing techniques in underdetermined linear models*, Ph.D. Thesis, nb. , Dalhousie University, Outubro de 1997.
- [24] BORMAN, S. e STEVENSON, R. "Spatial resolution enhancement of low -resolution image sequences: A comprehensive review with directions for future research", *SR Review*, Julho 1998.
- [25] NEUMAIER A. "Solving ill-conditioned and singular linear systems: A tutorial on regularization", *SIAM Review*, 1998.

Artigos de Conferências

- [26] SHIN, J. H., YOON, J. S., PAIK, J. K. e ABIDI, M. “Fast superresolution for image sequences using motion adaptive relaxation parameters”, *Proc. 1999 Int. Conf. Image Processing*, vol. 3, p. 676-680, Outubro de 1999.
- [27] HONG, M. C., KANG, M. G. e KATSAGGELOS, A. K. “An iterative weighted regularized algorithm for improving the resolution of video sequences”, *Proc. 1997 Int. Conf. Image Processing*, vol. 2, p. 474-477, Outubro 1997.

Arquivos de Ajuda de Softwares

- [28] MATLAB, 1999. Versão 5.3. Aplicativo para simulações de algoritmos matemáticos em Sistemas Operacionais Windows. *The MathWorks, Inc.*

Indicações Sobre Referências Encontradas

A maioria das referências foi conseguida realizando *download* da internet, através dos *sites* de periódicos da CAPES (<http://www.periodicos.capes.gov.br>), pelo *site do NEC Research Institute CiteSeer* (<http://citeseer.nj.nec.com>), ou dos *sites* dos próprios autores.

Glossário

<i>A Priori</i>	previamente, anterior.
<i>Affine</i>	tipo de movimento, descrito por seis parâmetros, ver Tabela 3.1.
<i>Aliasing</i>	efeito de sobreposição de espectro durante os processo de amostragem ou subamostragem.
<i>Average Power</i>	potência média.
<i>Blurring</i>	distorção óptica com aparência de vista borrada.
<i>Bounding</i>	limitação, contorno, circunscrição.
<i>Constrained</i>	restrito, restringido.
<i>Displacement Vector</i>	vetor de deslocamento, usado para representar o deslocamento de um <i>pixel</i> de um quadro para o outro.
<i>Download</i>	Processo de transferência de arquivos de servidores na <i>internet</i> para computadores pessoais.
<i>Downsampling</i>	processo de redução da taxa de amostragem.
<i>Frame</i>	imagem, tela ou quadro individual de uma seqüência de imagens ou vídeo.
<i>High-Resolution</i>	alta resolução, ou resolução suficiente ou superior a necessária para representar um sinal contínuo através de amostras.
<i>Ill-Conditioned</i>	mal-condicionada, condição de matrizes ou sistemas lineares.
<i>Inliers</i>	regiões que são visíveis em todos os quadros de uma seqüência de imagens.
<i>Intensity Matching</i>	combinação de intensidade, técnica que busca intensidades de amplitudes iguais ou mais próximas possíveis.
<i>Low-Resolution</i>	baixa resolução, ou resolução insuficiente para representar a imagem contínua através de amostras.
<i>Motion Blur</i>	distorção óptica causada pelo movimento, com aparência de borrado tipo rastro.
<i>Outliers</i>	regiões que são visíveis em um quadro da seqüência de imagens, mas não são visíveis nos demais quadros.
<i>Pixel</i>	ponto ou amostra de uma imagem digital.
<i>Rank</i>	posto de uma matriz, numero de valores singulares maiores que zero.
<i>Regularized</i>	regularizado, estabilizado.

<i>Simultaneous Displacement</i>	deslocamento simultâneo, técnica usada em busca de soluções, ver Capítulo 5.
<i>Steepest Descent</i>	descida mais íngreme, técnica usada em busca de soluções para equações, ver Capítulo 5.
<i>Subpixel</i>	fração de um <i>pixel</i> .
<i>Upsampling</i>	processo de aumento da taxa de amostragem.